Data Description Guide

CRSP US Stock and CRSP US Indices Databases

105 West Adams Street, Suite 1700
Chicago, IL 60603
Tel: 312.263.6400
Fax: 312.263.6430
Email: Support@crsp.ChicagoBooth.edu

Version CA297.200809

CHAPTER 1: INTRODUCTION

1.1 About CRSP

The Center for Research in Security Prices (Prof. Eugene F. Fama, Chairman) has been an integral part of the academic and commercial world of financial and economic research. Since its inception in 1960, CRSP has provided an unparalleled foundation as the leading source for the most comprehensive and accurate historical US databases available. CRSP is a research institute of the Graduate School of Business of the University of Chicago, which has a history of being a catalyst for innovation and progress, and has been a resource for other academic institutions and corporations alike.

In 1959, Louis Engel, vice president of Merrill Lynch, Pierce, Fenner & Smith, called Professor James H. Lorie (PhD 1947; Professor of Business Administration) with an inquiry which resulted in a grant from Merrill Lynch and the establishment of CRSP.

The inquiry developed into a project which involved compiling, cleaning and codifying the prices, dividends and rates of return of all stocks listed and trading on the NYSE since 1926. It resulted in an academic research-grade database that remains invaluable to empirical research due to its breadth, depth, and completeness, and includes CRSP's unique permanent identifiers, allowing for clean and accurate time-series research and event studies.

CRSP files continue to provide a strong foundation for economic forecasting, stock market research, and financial analyses by academic institutions, investment banks, brokerage firms, corporations, banks and government agencies. CRSP provides the following data files: common stocks on the NYSE, AMEX, NASDAQ and NYSE Arca; CRSP Indices; NASDAQ, and S&P 500 composite indices; NASDAQ and AMEX Industry Indices; US Treasury bonds; Survivor-Bias-Free Mutual Funds; market capitalization reports; proxy graphs for 10K SEC filings and custom datasets. Additionally, CRSP continues to develop new research resources such as the new CRSP/Ziman Real Estate Data Series.

1.2 Development of the CRSP Stock Files

CRSP Stock File Data Dates By Exchange

Exchange	Monthly Stock Files Beginning Date	Daily Stock Files Beginning Date
NYSE	12/31/1925	12/31/1925
AMEX	07/02/1962	07/02/1962
NASDAQ	12/29/1972	12/14/1972
NYSE ARCA	03/31/2006	03/08/2006

The CRSP Data Files were developed by the Center for Research in Security Prices (CRSP), Graduate School of Business, University of Chicago. Lawrence Fisher, currently at Rutgers University, and James Lorie, Professor Emeritus of Business Administration in the Graduate School of Business at the University of Chicago, built the CRSP stock file and originated its basic design and content. For a more complete discussion of the original files, see Lawrence Fisher and James H. Lorie, *A Half Century of Returns on Stocks and Bonds,* Chicago: The University of Chicago, Graduate School of Business, 1977, Appendices A and B.

The original CRSP stock file contained month-end prices and returns from the New York Stock Exchange (NYSE) dating from December, 1925. Daily and monthly American Stock Exchange (AMEX) data and Daily NYSE data beginning in July, 1962, were respectively combined into monthly and daily NYSE/AMEX files providing price and return information on NYSE/AMEX common stock securities. CRSP's data coverage expanded in 1987 to include NASDAQ daily and monthly stock data, with information for domestic common stocks and ADRs traded on the NASDAQ Stock Market beginning December 14, 1972. NYSE, AMEX, and NASDAQ security data are available merged in the Daily or Monthly CRSP US Stock Database.

In 2005, CRSP completed the compilation and merging of daily data between 1925 and 1962 for securities that traded on NYSE in that period, resulting in seamless daily and monthly data for securities trading on NYSE between as early as December 1925.

NYSE Arca daily and monthly data were added in July 2007 for securities with primary listings on that exchange. NYSE Arca coverage begins on March 8, 2006.

NASDAQ Markets

The NASDAQ Stock Market[SM] consists of two subsets of securities, The NASDAQ National Market and The NASDAQ SmallCap Market. Currently, for a security to be designated a NASDAQ National Market Security, it must meet criteria setting minimum levels for: annual income, numbers of publicly traded shares, market capitalization, share price, and number of market makers. All other securities belong to The NASDAQ SmallCap Market. A security may move between The NASDAQ National Market and The NASDAQ SmallCap Market over time as its status changes.

The NASDAQ National Market was initiated in April 1982 for larger and generally more actively traded NASDAQ securities. The NASDAQ National Market Securities must meet higher financial and non-financial criteria than other NASDAQ stocks, and are subject to last-sale reporting. In June of 1992 the regular NASDAQ segment of The NASDAQ Stock Market[SM] was renamed The NASDAQ SmallCap Market and for the first time these became subject to real-time price and volume reporting.

The CRSPAccess NASDAQ security data includes the closing bid, closing ask, and the number of trades, formerly included in the CRSP Supplemental NASDAQ Data File. The latter data items have been reported for issues listed on The NASDAQ National Market since November 1, 1982. Issues listed on The NASDAQ SmallCap Market have had these data reported since June 15, 1992.

For a more detailed description of how to identify The NASDAQ National Market and The NASDAQ SmallCap Market securities, see The NASDAQ National Market indicator definition in the NASDAQ information history array described in the data definitions section.

Data Accuracy of the CRSP Stock Data

CRSP stock files are designed for research and educational use and have proven to be highly accurate. Considerable resources are expended in ongoing efforts to check and improve data quality both historically and in the current update. Data corrections to historical information are made as errors are identified and are detailed in the release notes that accompany each data cut.

Historical corrections to security data may result in changes to historical CRSP index returns and levels. In any given year, the calendar year-end stock database may ship as the standard "Z-Cut", as well as a second "X-Cut" if there have been substantial or significant corrections applied at year end. Both data cuts reflect corrections, however, the "Z-Cut" contains routine minor monthly corrections while the "X-Cut's" year-end data edits typically required more extensive research efforts and may hold a greater potential to impact the CRSP indices.

Data Sources

Original CRSP Monthly Database

- December 1925-January 1928: *Commercial and Financial Chronicle*, "Bank and Quotation Section"
- February 1928-December 1960: *Bank and Quotation Record*, an expansion of the Bank and Quotation Section.

The collection and initial correction of cash dividends in the original CRSP Monthly Database was performed as follows:

- 1937-1960: annual issues of *Moody's, Standard and Poor's Dividend Records*, or the annual section of the *Standard Corporation Records* were used, depending on which was in the University of Chicago Library
- 1926-1936: *Moody's Quarterly Dividend Record*

  Since the only known complete file of this last publication was in Moody's New York offices, the data from the earlier period were recorded in the Moody's offices by trainees working for the research division of Merrill Lynch, Pierce, Fenner & Smith Inc.

Pre62 Database

The primary source for the Pre62 daily was The New York Times newspaper. In cases where the stock information was either missing from The New York Times or the available data were questionable, The Wall Street Journal Newspaper served as CRSP's secondary data source. For the years prior to 1933, CRSP compared The New York Times and The Wall Street Journal data for distributions with the original database. CRSP determined that The New York Times and The Wall Street Journal ex-date data were more complete and accurate than the original data. Based on these findings, CRSP made the decision to rely on The New York Times as the primary data source for distribution ex-date data from 1925 through 1933.

CRSP US Stock & Indices Database

NYSE and AMEX

- July 1962-September 1, 1972: daily price and dividend data provided by Standard & Poor's Price Tape and Punched Card Dividend Service
- July 1962-March 1987: High, low, and volume data provided by Interactive Data Services, Inc. (IDSI), a subsidiary of Interactive Data Corporation (IDC)
- September 1972-April 1987: Interactive Data Corporation (IDC)

  The Standard & Poor's Price Tape and Punched Card Dividend Service was acquired by IDC.
- April 1987-September 1999: Interactive Data Services, Inc. (IDSI)
- 1999-present: Interactive Data Corporation

NYSE Arca

- March 2006-present: Interactive Data Corporation (IDC)

  Coverage of companies with primary listings on NYSE Arca who have traded since 3/8/2006.

NASDAQ

- December 12, 1972-August 31, 1984: Interactive Data Corporation (IDC)
- November 1, 1982-present (with the exception of February 1986): National Association of Securities Dealers (NASD)
- November 1, 1982-August 31, 1984: Interactive Data Corporation (IDC) was used as a secondary source to NASD
- February 1986: Interactive Data Services, Inc. (IDSI)
- March 2004-present: Interactive Data Corporation used as secondary source

CHAPTER 2: CRSP DATABASE STRUCTURE

This section contains general and background information about the organization and content of the CRSP stock and indices databases. Diagrams are based on C and FORTRAN access of the databases.

There are four types of CRSP data available:

1. Header identification or summary data valid for the entire range of the security or index.
2. Time series data with observations recorded for each period in an associated calendar.
3. Event data representing status changes, random events, or observations. The time of the event and relevant information is stored as different elements of the event.
4. Calendar data including the calendar periods used to synchronize time series observations. Daily databases are based on calendars of daily trading dates, and monthly databases are based on calendars of month-end trading dates.

CRSP stock data are organized by security. Index data are organized either by index series or collections of index series.

CRSP Calendars

A CRSP calendar is a set of time periods with header information about those time periods. The calendar time periods are chosen as points of interest rather than all calendar days, and therefore a daily calendar contains only the dates where trading was conducted on a major US exchange. Data are never provided on weekends or trading holidays. The standard identification of a time period is the date, an integer in YYYYMMDD format, at the end of the period.

There are currently five calendars provided with CRSPAccess databases: daily, monthly, weekly, quarterly, and annual. The daily calendar is used to derive the others so that the last trading date in each month, week, quarter, or year is used to build those respective calendars.

Time series data are always associated with one of these calendars. The list of time series observations is synchronized with a calendar so that the n^{th} time series observation is associated with the n^{th} calendar period.

A Calendar Name and an integer Calendar Identification Number identify each calendar. The calendars supported in CRSPAccess databases are:

Calendars	Calendar Identification Number	Calendar Name	Beginning Date
Daily	100	Daily Trading Calendar	19251231
Monthly	101	Month-end Trading Calendar	19251231
Annual	300	Annual Trading Calendar	19251231
Quarterly	310	Quarterly Trading Calendar	19251231
Weekly	500	Weekly Trading Calendar	19260102

Stock Data Universe

CRSP stock data includes data from NYSE, AMEX, NASDAQ, and NYSE Arca stock exchanges. The following items are included in our stock databases:

- Common Stocks
- Certificates
- ADRs
- Shares of Beneficial Interest
- Units (Depository Units, Units of Beneficial Interest, Units of Limited Partnership Interest, Depository Receipts, etc.
- Closed-End Mutual Funds
- Foreigns on NYSE, AMEX, NASDAQ and NYSE Arca
- Americus Trust Components (Primes and Scores)
- HOLDRs Trusts
- REITs (Real Estate Investment Trusts)

CRSP stock databases exclude:

- Rights and Warrants
- Preferreds
- Units Representing Common Stocks Bundled with Rights or Warrants
- Over the Counter Bulletin Board Issues
- When Issued Trading

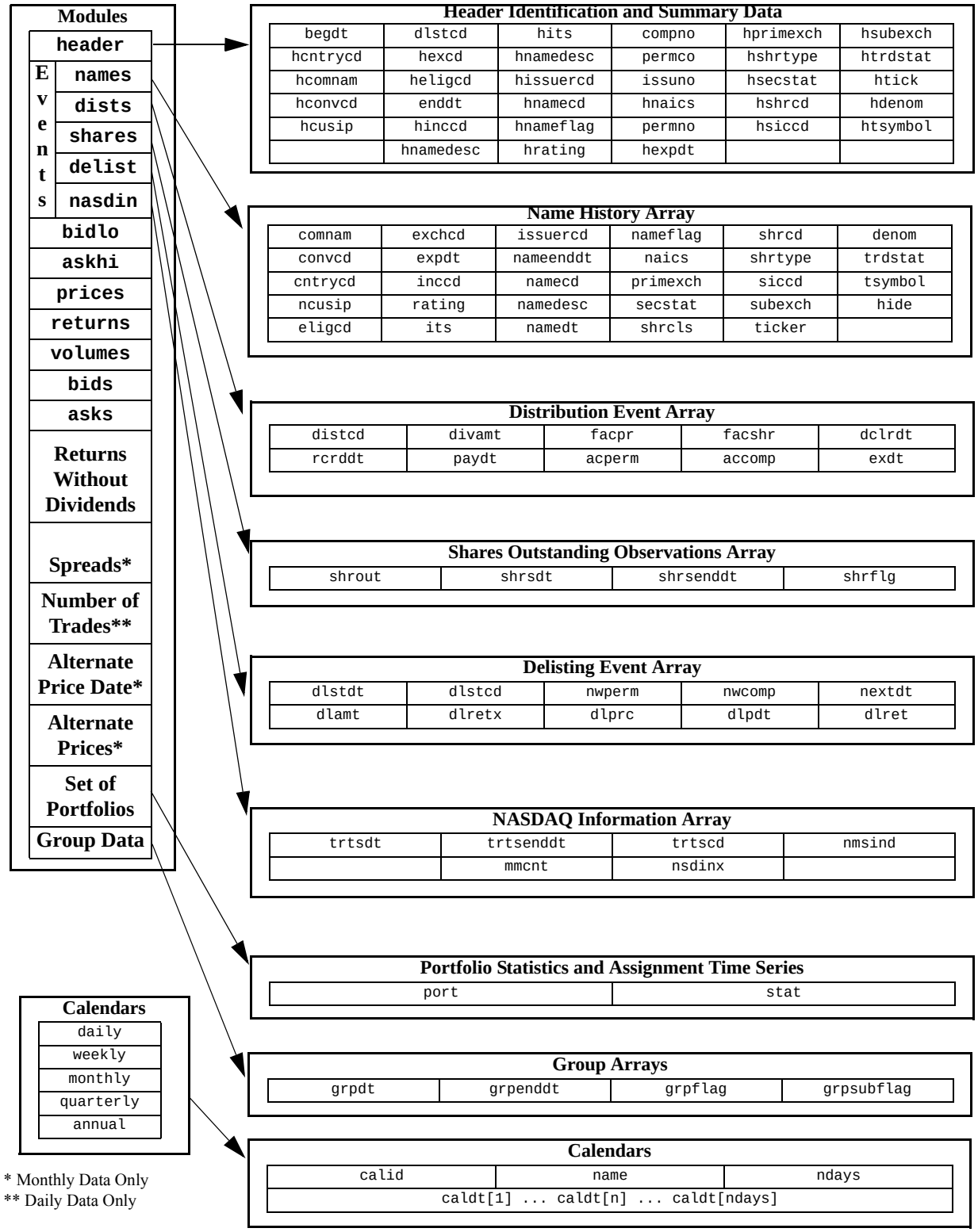
Database diagrams with the data variables are organized by format and programming use. The variables are organized by mnemonic in each diagram. Descriptions following the diagrams expand the major data structures and list the variable names of all data arrays and elements. The variables are defined in the Data Definitions section of this guide. See “Chapter 4: Data Definitions” on page 39.

There are two different database formats for the CRSP stock and indices data: CRSPAccess. The primary differences between them are the way indices and portfolio data are organized. The data organization diagrams show the major data structures for C and FORTRAN. See the Programmers Guide for details on programming use.

2.1 CRSP Database Structure Diagrams

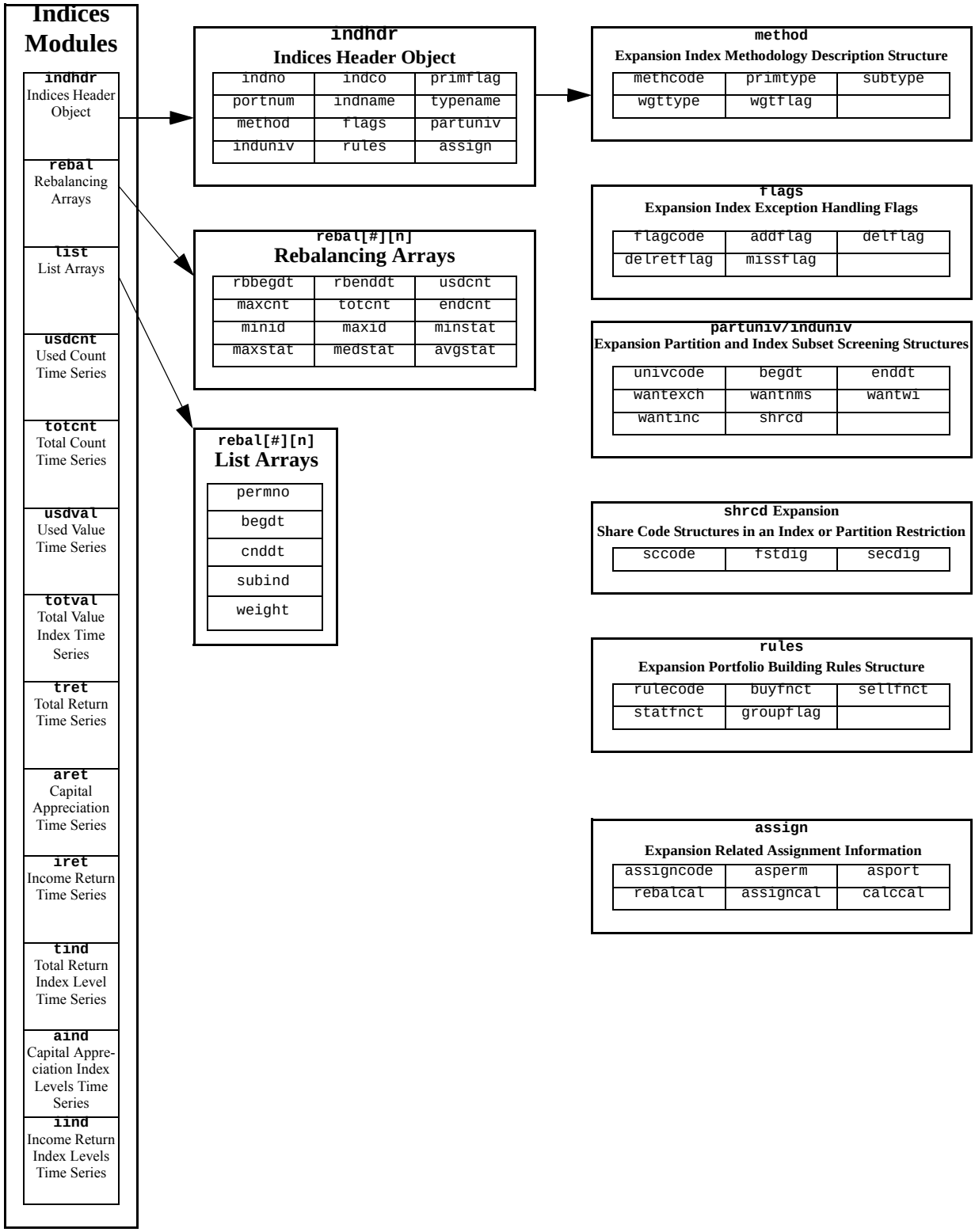
The database structures in CRSPAccess are organized by format and programming use. The variables are listed by mnemonic in each diagram. Section 2.4, Concepts, describes each array structure and includes the portion of the diagram that pertains to the description and has the name with the mnemonic. Both variable names and mnemonics are included in the index if you wish to cross reference them. (See “Chapter 4: Data Definitions” on page 39.) The Data Definitions chapter is organized alphabetically by variable name.

CRSP Stock Data Layout - CRSPAccess



* Monthly Data Only
** Daily Data Only

CRSP Indices Data Layout - CRSPAccess



2.2 Stock and Indices Data Structures

Stock Data Structures

This section describes each stock data array included in the CRSPAccess database. Accessibility and data structures between C and FORTRAN are only slightly different. When they differ, a description and layout is included for each. Please refer to the diagrams in Section 2.1 to determine which of the following arrays pertain to your usage.

Header Identification and Summary Data

Header Identification and Summary Data is a set of variables, in a CRSPAccess stock database using CRSP C access functions, that identify an issue and summarize its classification. There is no time component to the header data so the data are valid the entire range of the issue. Header Identification and Summary Data contains the most current information on the issue maintained in the file. There is only one header structure per issue for any data iteration. Note that Ticker Symbol header only contains tickers for active securities.

Header Identification and Summary Data - `header`

	Variable Name	Variable
Primary Permanent Identifiers	PERMCO	permco
	PERMNO	permno
Secondary Permanent Identifiers	CUSIP - Header	hcusip
	NASDAQ Company Number	compno
	NASDAQ Issue Number	issuno
Security Date Ranges	Begin of Stock Data	begdt
	End of Stock Data	enddt
Most Recent Header Identification and Summary Data Information	Company Name - Header	hcomnam
	Convertible Code - Header	hconvcd
	Country Code - Header	hcntrycd
	Eligibility Code - Header	heligcd
	Exchange Code - Header	hexcd
	Expiration Date	hexpdt
	Incorporation Code - Header	hinccd
	Interest Rate or Strike Price	hrating
	Intermarket Trading System Indicator - Header	hits
	Issue Description - Header	hnamedesc
	Issuer Code - Header	hissuercd
	Name Code - Header	hnamecd
	Name Description - Header	hnamedesc
	Name Flag - Header	hnameflag
	North American Industry Classification Code - Header	hnaics
	Primary Exchange - Header	hprimexch
	Security Status - Header	hsecstat
	Share Code - Header	hshrcd
	Share Type - Header	hshrtype
	Standard Industrial Classification (SIC) Code - Header	hsiccd
	Sub-exchange - Header	hsubexch
	Ticker Symbol - Header (active securities only)	htick
	Trading Denomination - Header	hdenom
	Trading Ticker Symbol - Header	htsymbol
Most Recent Listing Information	Delisting Code - Header	dlstcd
	Trading Status - Header	htrdstat

Name History Array

The Name History Array includes sets of identification variables effective at different times during the history of a security. Each set of information, or name structure, contains name and classification fields and the effective date ranges of those fields. Each security has at least one name structure.

Name History Array - names

	Variable Name	Variable
Secondary Identifiers	CUSIP	ncusip
	North American Industry Classification System (NAICS) Code	naics
	Ticker Symbol	ticker
	Standard Industrial Classification (SIC) Code	siccd
Date Range of Name History Record	Name Effective Date	namedt
	Last Date of Name[1]	nameenddt
Identifying Information	Company Name	comnam
	Convertible Code	convcd
	Country Code	cntrycd
	Eligibility Code	eligcd
	Exchange Code	exchcd
	Expiration Date	expdt
	Incorporation Code	inccd
	Interest Rate or Stricke Price	rating
	Intermarket Trading System Indicator	its
	Issuer Code	issuercd
	Name Code	namecd
	Name Description	namedesc
	Name Flag	nameflag
	Primary Exchange	primexch
	Security Status	secstat
	Share Class	shrcls
	Share Code	shrcd
	Share Type	shrtype
	Sub-exchange	subexch
	Trading Denomination	denom
	Trading Status	trdstat
	Trading Ticker Symbol	tsymbol

[1]CRSPAccess only

If the CUSIP, Company Name, Exchange Code, Exchange Ticker Symbol, Share Class, or SIC Code changes during the security's trading history, a new name structure is added, with the Name Effective Date of the change. That information is valid until another name structure is added or the security becomes obsolete.

Name Histories may include periods, possibly outside the data range, when the security is trading on a different exchange or is not trading at all. The Exchange Code description (Page 59) contains more detailed information on trading status and location for a given date range.

Distribution Event Array

The Distribution Event Array is a list of events describing cash dividends, capital adjustments, and other distributions made to shareholders of a security.

Distribution Event Array - `dists`

	Variable Name	Variable
Distribution Information	Distribution Code	distcd
	Dividend Cash Amount	divamt
Factors to Adjust Prices and Shares	Factor to Adjust Price	facpr
	Factor to Adjust Shares Outstanding	facshr
Dates Associated with the Distribution	Distribution Declaration Date	dclrdt
	Ex-Distribution Date	exdt
	Record Date	rcrddt
	Payment Date	paydt
Securities/Companies Related to the Event	Acquiring PERMNO[1]	acperm
	Acquiring PERMCO[1]	accomp

[1]CRSPAccess variable only, available in both FORTRAN and C.

If a distribution event has more than one component, CRSP codes each component of the event separately with a four-digit code. All components of a distribution event share the same Ex-Distribution Date. Distributions for each security are unique and are sorted by Ex-Distribution Date, Distribution Code, and Acquiring PERMNO.

Distribution Events are a descriptive set of events, not a summary by period. The data can be summarized for returns calculations, delisting returns, price and shares adjustments, and dividend and split totals. The following types of event are available:

- Periodic and special cash dividends – the cash amount in US dollars, frequency, and related dates of all cash dividends are provided.
- Stock splits, stock dividends, and reverse splits – the factors to adjust price and shares, type of action, and related dates of all splits are provided.
- Spin-offs – All spin-off events are included. The cash value of the spin-off is the price at the end of the ex-distribution date of the stock received. A price factor is calculated by dividing the cash amount by the price of the parent security on the Ex-Distribution Date. Acquiring PERMNO and Acquiring PERMCO can be used to link to the new company when available.
- Liquidation payments – All partial and final liquidation payments are included. These contain the value of each payment and relevant dates that are known. If the payment is in the form of stock, or if a payment is known to come from the purchase of assets by a known company, the Acquiring PERMNO and Acquiring PERMCO are set to that company or issue.
- Return of capital distributions.
- Rights offerings.
- Merger, acquisition, and reorganization distributions.
- Limited tender offers.
- Information on announcements related to liquidations and tender offers that resulted in delistings.
- Known shares buybacks, offerings, and share increases due to acquisitions.

See “6.2 Distribution Codes” on page 126 for the coding scheme used by CRSP. See the table on page 138 for examples of specific cases of distributions.

Shares Outstanding Observations Array

The Shares Outstanding Observations Array contains the history of shares outstanding observations of a security. CRSP records the shares outstanding only for the security, not the total shares for the company. Treasury shares are not included. Shares outstanding for American Depositary Receipts (ADRs) are the shares outstanding of the ADR, not the underlying issue. Shares outstanding are recorded in thousands.

Shares Outstanding Observation Array - `shares`

	Variable Name	Variable
Shares Information	Shares Outstanding	`shrout`
	Shares Outstanding Observation Flag	`shrflg`
Share Observation Date Range	Shares Outstanding Observation End Date[1]	`shrsenddt`
	Shares Outstanding Observation Date	`shrsdt`

[1] CRSPAccess data access only.

There are two types of Shares Outstanding Observations:

1. Primary shares observations contain a shares outstanding amount taken directly from an annual or quarterly report or a data source using company reports.
2. These are supplemented with imputed shares observations derived from distributions affecting shares outstanding using Factor to Adjust Shares.

A new entry does not imply that there was a change in the number of shares outstanding. In general, every company has at least one shares structure per year.

Exactly one shares structure is effective each date in the security's history. One shares outstanding observation is effective until the next observation or the delisting date. The first shares observation is effective from the Shares Observation Date backward to the beginning of data.

The Shares Outstanding Observations Array cannot be used to directly find the shares outstanding each calendar period. Utility functions and programs are available to map observations to time series used to calculate market capitalization.

Delisting Event Array

Every security on the CRSP file is assigned one delist record. The Delisting Event Array contains information on the status of a security after it is no longer listed on an exchange in a CRSP file. Each delisting history event contains a code describing the reason for delisting, the value after delisting (when available), forward links to acquiring issue and company traded on NYSE, AMEX, NASDAQ, or NYSE Arca and delisting return. Active issues have a delisting history event where Delisting Date is set to the last date of available price data. The Distribution History Array includes itemized data on the payments made to shareholders after the delisting, and includes announcement information related to the delisting when available.

Delisting Event Array - `delist`

	Variable Name	Variable
Delist Information	Amount After Delisting	`dlamt`
	Delisting Code	`dlstcd`
	Delisting Price	`dlprc`
	Delisting Return	`dlret`
	Delisting Return without Dividends	`dlretx`
Dates Assocaited with Delist	Delisting Date	`dlstdt`
	Date of Next Available Information	`nextdt`
	Delisting Payment Date	`dlpdt`
Securities/Companies Associated with Delist	New PERMCO[1]	`nwcomp`
	New PERMNO[1]	`nwperm`

[1]CRSPAccess C and FORTRAN only.

In current CRSP files only the most recent delisting event is coded in the Delisting Event Array. If an issue leaves an exchange in the CRSP data files and later returns, the gap is marked in the Name History Array with an Exchange Code of 0. During this time, event data are not tracked and time series data are filled in with missing values.

Delisting information is determined by several factors: the exchange of shares at the earliest possible opportunity, trade on a secondary market, payments from the company, or outstanding tender offer. The information is coded as it becomes available. An issue is considered closed to further research if any of the following conditions apply:

- Research has verified that a final distribution has been paid to stockholders.
- A price is found on another exchange.
- Research has verified that no distributions were ever paid to stockholders.
- Some distributions have been paid to stockholders, but no final distribution information can be found and 10 years have passed since the date of the most recent delisting information.
- No information concerning the delisting can be found and 10 years have passed since the delist date.

If none of these conditions applies to a delisted issue, the issue is pending, which means that further research is required until one of the above conditions has been met. If no information is found or the information found is incomplete, no delisting return will be calculated by CRSP.

Monthly: If no delisting information exists, and daily data exist after the last month-end trading date, CRSP generates partial-month delisting amounts and returns by using the price on the last daily trading date. *Although the partial-month returns are stored in the Delisting Return field, they are not delisting returns.*

NASDAQ Information Array

The NASDAQ Information Array contains a history of an issue's trading status on The NASDAQ Stock Market[SM]. Each set of information, or structure, contains status and classification fields and the effective date ranges of those fields. If the NASDAQ Traits Code, NASDAQ National Market Indicator, NASD Index Code, or Market Makers Count changes, then a new structure is added, and the date of the change is recorded in the NASDAQ Traits Date. Each issue traded on the NASDAQ Stock Market since November 1982 has at least one NASDAQ Information Array.

NASDAQ Information Array - `nasdin`

	Variable Name	Variable
NASDAQ Information Array Data	Market Maker Count	`mmcnt`
	NASD Index Code	`nsdinx`
	NASDAQ National Market Indicator	`nmsind`
	NASDAQ Traits Code	`trtscd`
NASDAQ Information Date Range	NASDAQ Traits Date	`trtsdt`
	NASDAQ Traits End Date	`trtsenddt`

NASDAQ information structures are available for securities trading on NASDAQ beginning on April 1, 1982 for NASDAQ Traits Date and NASDAQ National Market Indicator. All fields are available beginning November 1, 1982[1].

The NASDAQ National Market was initiated in April 1982 for larger and generally more actively traded NASDAQ securities. The NASDAQ National Market Securities must meet higher financial and non-financial criteria than other NASDAQ stocks, and were always subject to last-sale reporting. In June of 1992 the regular NASDAQ segment of The NASDAQ Stock Market[SM] was renamed The NASDAQ SmallCap Market and for the first time these issues became subject to real-time price and volume reporting.

[1] NASDAQ information data are missing in December, 1982 for all issues with NASD company numbers less than 1025 (approximately 20 percent of the securities active at that time), and are missing in February, 1986 for all issues. NASDAQ Traits Date, NASDAQ Traits Code, and NASDAQ National Market Indicator are complete. All other fields are missing.

Price, Volume, and Return Time Series Arrays

Price, Volume, and Return Time Series Arrays are a set of time series that makes up the core of CRSP stock data. This includes three price time series, total returns, and trading volumes. All these time series in a stock file use the same calendar(s).

Primary Price, Volume, and Return Time Series Arrays

	Variable Name	Variable
Price, Volume, and Return Time Series Data	Ask or High Price	askhi
	Bid or Low Price	bidlo
	Holding Period Total Return	ret
	Price or Bid/Ask Average	prc
	Volume Traded	vol

Auxiliary Time Series Data

Auxiliary Data Time Series are additional time series provided in CRSPAccess stock files at the same frequency as the Price, Volume, and Return Time Series Arrays.

Auxiliary Time Series Data

	Variable Name	Variable
Monthly Alternate Price and Date, Daily Open Price, Returns Without Dividends, Spread Time Series Data	Price Alternate (monthly only)	altprc
	Price Alternate Date (monthly only)	altprcdt
	Return Without Dividends	retx
	Spread Between Bid and Ask (monthly only)	spread
	Open Price (daily only)	openprc
	Ask	ask
	Bid	bid
	NASDAQ Number of Trades (daily only)	numtrd

Portfolio Statistics and Assignment Time Series

Portfolio Statistics and Assignment Time Series is a set of portfolio time series. Each portfolio time series is based on a portfolio type defined by CRSP and contains a history of statistics and portfolio assignments for a security. Two variables are available for each calendar period:

Portfolio Statistics and Assignment Time Series - `port`

	Variable Name	Variable
Portfolio Statistic and Assignment Times Series Data	Portfolio Assignment Number	`port`
	Portfolio Statistic Value	`stat`

Each Portfolio Statistics and Assignment Time Series in the set is called a Portfolio Type. Portfolio Types are predefined groupings based on CRSP indices. The portfolio time series can be linked to CRSP index returns data to calculate excess returns of a security against its assigned index portfolio at any time during its history.

Each Portfolio Type represents a predefined index group with its own methodology and rebalancing period. The portfolio time series can be linked to different calendars based on the rebalancing frequency of the index. The timing and calculation of the statistic and assignment rules are also dependent on the index. Calendars used in portfolios are not the same calendars used with security price and returns data. Portfolio ranges and calendars can differ for all portfolio types. In a portfolio time series, the Data Subtype Code is set to the Permanent Index Identification Number, INDNO, of an index that contains the performance results of the group of index series built using the assignments.

The portfolio assignments for the CRSP Stock File Decile Capitalization Indices for NYSE/AMEX/NASDAQ are provided with daily and monthly stock files. Additional Portfolio Types are available with the CRSP US Indices Database and Security Portfolio Assignment Module. Note that the portfolio information is a module of the associated CRSPAccess daily or monthly stock data. Portfolio assignment data for daily or monthly indices can be extracted through the stock utilities when the user subscribes to the appropriate stock and indices products. Indices based on the portfolios are included in the CRSP Indices File and Portfolio Assignments product.

See the Portfolio Type table (Page 37) in the Index Methodologies section for more details about the defined portfolios available in monthly and daily stock files.

Group Data

Group Data are a set of arrays of universe inclusion events. Each supported universe is called a `grouptype` and assigned an integer number that identifies it. The array for each `grouptype` lists the number of universe events and dates.

Group Data

	Variable Name	Variable
Group Data	Group Flag of Associated Index	`grpflag`
	Group Secondary Flag	`grpsubflag`
Group Date Range	Begin of Group Data	`grpdt`
	End of Group Data	`grpenddt`

The only `grouptype` currently available is 16 - S&P 500 Universe. Only inclusion events are added to this group, so Group Flag of Associated Index is always 1. Begin of Group Data and End of Group Data identify the range the security was included in the S&P 500 index. The user must subscribe to the appropriate stock and indices databases to extract group data.

CRSPAccess Index Data Structures

There are four types of index data provided with the CRSPAccess indices data; header, rebalancing, list and time series. Additionally, security portfolio assignment data are provided in association with market segment portfolio groups. Daily and monthly indices can be integrated with CRSPAccess stock files to provide excess returns on the fly.

Index Header

The Index Header is a set of fields containing identification and methodology information about an index series or group. See "Chapter 3: CRSP Index Methodologies" on page 27 for more descriptive information about the methodologies of the CRSP index types.

Index Header - `indhdr`

	Variable Name	Variable
Permanent Index Identifiers	INDNO	indno
	INDCO	indco
Descriptive Identifiers	Index Primary Link	primflag
	Portfolio Number if Subset Series	portnum
	Index Name	indname
	Index Group Name	groupname
Index Structures (detailed below)	Index Methodology Description Structure	method
	Index Exception Handling Flags	flags
	Index Subset Screening Structure	partuniv
	Partition Subset Screening Structure	induniv
	Portfolio Building Rules Structure	rules
	Related Assignment Information	assign

Index Methodology Description Structure - `method` - Expansion

	Variable Name	Variable
Methodology Description Structure	Index Method Type Code	methcode
	Index Primary Methodology Type	primtype
	Index Secondary Methodology Group	subtype
	Index Reweighting Type Flag	wgttype
	Index Reweighting Timing Flag	wgtflag

Index Exception Handling Flags - `flags` - Expansion

	Variable Name	Variable
Exception Handling Flag Structure	Index Basic Exception Types Code	flagcode
	Index New Issues Flag	addflag
	Index Ineligible Issues Flag	delflag
	Return of Delisted Issues Flag	delretflag
	Index Missing Data Flag	missflag

Partition/Index Subset Screening Structure - `partuniv/induniv` - Expansion

	Variable Name	Variable
Index Subset Screening Structure and Partition Subset Screening Structure	Universe Subset Types Code in a Restriction	`univcode`
	Restriction Beginning Date	`begdt`
	Restriction End Date	`enddt`
	Valid Exchange Codes in Universe in a Restriction	`wantexch`
	Valid NASDAQ Market Groups in Universe in a Restriction	`wantnms`
	Valid When-Issued Securities in Universe in a Restriction	`wantwi`
	Valid Incorporation of Securities in Universe in a Restriction	`wantinc`
	Share Code Screen Structure in a Restriction	`shrcd`

Share Code Screen Structure in a Partition or Index Restriction - `shrcd` - Expansion

	Variable Name	Variable
Share Code Screen Structure	Share Code Groupings for Subsets in a Restriction	`sccode`
	Valid First Digit of Share Code in a Restriction	`fstdig`
	Valid Second Digit of Share Code in a Restriction	`secdig`

Portfolio Building Rules Structure - `rules` - Expansion

	Variable Name	Variable
Portfolio Building Rules	Index Basic Rule Types Code	`rulecode`
	Index Function Code for Buy Rules	`buyfnct`
	Index Function Code for Sell Rules	`sellfnct`
	Index Function Code for Generating Statistics	`statfnct`
	Index Statistic Grouping Code	`groupflag`

Portfolio Building Rules Structure - `assign` - Expansion

	Variable Name	Variable
Portfolio Building Rules Structure	Index Basic Assignment Types Code	`assigncode`
	INDNO of Associated Index	`asperm`
	Portfolio Number in Associated Index	`asport`
	Calendar Identification Number of Rebalancing Calendar	`rebalcal`
	Calendar Identification Number of Assignment Calendar	`assigncal`
	Calendar Identification Number of Calculations Calendar	`calccal`

Index Rebalancing History Arrays

The Index Rebalancing History Arrayss are a set of CRSPAccess event array structures containing decile-level historical rebalancing statistical information for rebalancing periods in an index. Each event array structure within the history contains the characteristics for one portfolio for one time range in the index, including the breakpoints used to assign securities to the portfolio.

Index Rebalancing History Arrays - `rebal[#][n]`

	Variable Name	Variable
Rebalancing Date Ranges	Index Rebalancing Begin Date	rbbegdt
	Index Rebalancing End Date	rbenddt
Rebalancing Portfolio Statistics	Count Used as of Rebalancing	usdcnt
	Maximum Count During Period	maxcnt
	Count Available as of Rebalancing	totcnt
	Count at End of Rebalancing Period	endcnt
Breakpoint Information	Statistic Minimum Identifier	minid
	Statistic Maximum Identifier	maxid
	Statistic Minimum in Period	minstat
	Statistic Maximum in Period	maxstat
	Statistic Median in Period	medstat
	Statistic Average in Period	avgstat

Not all statistics are available for each index.

The variable Number of Rebalancing Types contains the count of the rebalancing history arrays available for all indices in a set. There are ten possible rebalancing arrays in current Index Groups and one in all Index Series. Each array has its own count of periods, which is set to zero if not applicable to the particular index.

Index List History Array

Index List History Arrays are a set of CRSPAccess event array structures containing lists of issues constituting an index.

Index List History Array - `list[#][n]`

	Variable Name	Variable
Security Identifier	Permanent Number of Securities in Index List	permno
Date Range	First Date Included in List	begdt
	Last Date Included in List	enddt
Security Characteristics	Index Subcategory Code	subind
	Weight of Issue	weight

The variable number of List Types contains the count of issue lists available for all indices in a set. There is one possible list array in Index Groups and Index Series. Each array has its own count, which is set to zero if not applicable to the particular index.

No list histories are available in the current index database.

Index Time Series

Index Time Series are sets of result and summary time series arrays for indices. They include the following variables:

Index Time Series

	Variable Name	Variable
Index Summary Statistics	Index Used Count	usdcnt
	Index Total Count	totcnt
	Index Used Value	usdval
	Index Total Value	totval
Index Returns	Index Total Return	tret
	Index Capital Appreciation Return	aret
	Index Income Return	iret
Index Levels	Index Total Return Index Level	tind
	Index Capital Appreciation Index Level	aind
	Index Income Return Index Level	iind

The variable Number of Index Types contains the count of index series available for each of the indices in a set. There is always one time series for all data items in an index series, and more than one time series for data items in an index group. Not all time series are available for each index. If the range for one of the time series is not set, data of that type is not available for that index.

2.3 Base CRSPAccess Data Structures

Base Data Structures

Time Series Objects

Time Series Objects are data structures used to store time series data in CRSPAccess databases. A CRSP time series contains information about the type of data stored for each observation, the ranges of valid data for the current entity, the actual list of data observations, and the calendar information needed to place the observations in time.

Time Series Objects

	Variable Name	Variable
Data Description Information	Object Type Code	objtype
	Array Type Code	arrtype
	Data Subtype Code	subtype
	Array Structure Size	size_of_ array_width
	Maximum Number of Array Elements	maxarr
Ranges of Valid Data	Begin of Valid Data	beg
	End of Valid Data	end
Associated Calendar Information	Calendar Time Period Description Code	caltype
	Calendar Associated with Time Series	cal
Object Array	Object Array	arr

Event Array Objects

Event Array Objects are data structures used to store event data in CRSPAccess databases. A CRSP event array contains information about the type of data stored for each observation, the number of events for the current entity, and the actual event observations. The event dates or effective date ranges are contained within the observations.

Event Array Objects

	Variable Name	Variable
Data Description Information	Object Type Code	objtype
	Array Type Code	arrtype
	Data Subtype Code	subtype
	Array Structure Size	size_of_ array_width
	Data Secondary Subtype Code	dummy
	Maximum Number of Array Elements	maxarr
Number of Array Elements	Number of Array Elements	num
Object Array	Object Array	arr

Header Objects

Header Objects are data structures used to store header data in CRSPAccess databases. A CRSP Header Object contains information about the type of data stored and the actual header fields.

Header Objects

	Variable Name	Variable
Data Description Information	Object Type Code	objtype
	Array Type Code	arrtype
	Data Subtype Code	subtype
	Array Structure Size	size_of_ array_width
Object Array	Object Array	arr

Calendar Objects

Calendar Objects are data structures used to store calendar data in CRSPAccess databases. A Calendar Object contains information about the type of data stored, descriptive information about the calendar, the number of time periods available, and lists of calendar periods. The Calendar Objects are used with Time Series Objects to match data observations with a point in time. The calendar periods are usually identified by the last trading date in the period.

Calendar Objects

	Variable Name	Variable
Data Description Information	Object Type Code	objtype
	Maximum Number of Array Elements	maxarr
	Calendar Type Availability Flag	loadflag
Ranges of Valid Data	Number of Periods in Calendar	ndays
Calendar Description Information	Calendar Name	name
	Calendar Identification Number	calid
Calendar Period Arrays	Calendar Period Grouping Identifiers	callist
	Calendar Trading Date	caldt

CHAPTER 3: CRSP INDEX METHODOLOGIES

CRSP provides a wide range of indices that can be used as benchmarks of market performance. Broad market indices are provided with CRSP stock files. Additional market indices, stock portfolios, bond indices, and inflation series are provided with CRSP indices files. The combination of portfolio results and assignment data provided with CRSP indices files added to the security data in CRSP stock files allows a comparison of securities against comparative benchmarks with a historical perspective.

This section describes the general methodologies for each of the major index types CRSP provides. These are:

- CRSP Stock File Indices, including:
  - CRSP Market Indices,
  - Published S&P 500 and NASDAQ Composite Index Data,
  - CRSP Stock File Capitalization Decile Indices,
  - CRSP Stock File Risk-Based Decile Indices,
- CRSP Cap-Based Portfolios,
- CRSP Indices for the S&P 500 Universe, and
- CRSP US Treasury and Inflation Series.

Lists of all Index Series and Groups and Portfolio Types provided by CRSP are included at the end of this section.

See "Chapter 5: CRSP Calculations" on page 105 and see Appendix A: CRSP Terminology for more specific details about the calculations and definitions used.

Index Data Availability

Market index returns, levels, and basic market statistics are available in ASCII, Excel, and SAS files. Decile level statistics and portfolio assignments are implemented as add-on modules to the corresponding CRSPAccess stock data. They are accessible through the stock data.

3.1 Stock File Indices

The CRSP Stock File Indices are a set of Market Indices and Decile Portfolio Indices provided daily, monthly, quarterly, and annually for five market groups of securities. The market groups of securities for which indices are calculated are the individual NYSE, AMEX, NASDAQ and NYSE Arca markets, the NYSE/AMEX, NYSE/AMEX/NASDAQ, and the NYSE/AMEX/NASDAQ/NYSE Arca market combinations. Published S&P 500 and NASDAQ Composite Index Data are also included.

The ranges for individual exchange data are listed below. The series containing combinations of exchanges begin at the earliest point that data for any of the exchanges is available.

The New York Stock Exchange (NYSE)	all series	begins December 31, 1925
The American Stock Exchange (AMEX)	all series	begins July 2, 1962
The NASDAQ Stock Market (NASDAQ)	all series	begins December 14, 1972
The NYSE Arca Exchange (ARCA)	all series	begins March 8, 2006

NOTE: Quarterly and annual index returns are not available for the series including Arca.

Daily and monthly index returns are calculated based on daily and monthly security holding period returns respectively. Quarterly and annual frequency index returns are calculated by compounding monthly index returns.

CRSP Market Indices

An Equal-Weighted Index and a Value-Weighted Index are calculated for each market group. Each index contains index returns with and without dividends, index weights and counts.

The Equal-Weighted Index is an Equal-Weighted Portfolio built each calendar period from all issues listed on the selected exchanges with valid prices on the current and previous periods.

The Value-Weighted Index is a Value-Weighted Portfolio built each calendar period using all issues listed on the selected exchanges with available shares outstanding and valid prices in the current and previous periods, excluding American Depositary Receipts. Issues are weighted by their Market Capitalization at the end of the previous period.

An additional daily trade-only value-weighted index is available for NYSE/AMEX. This index uses the same methdology as the NYSE/AMEX Value-Weighted Market Index, but only includes non-ADR securities with trades on current and previous trading days.

Index Levels of CRSP Market Indices are set to 100.0 on December 29, 1972.

The NYSE/AMEX/NASDAQ Market Indices are available in Daily and Monthly Stock Files. Other exchange combinations are available in the CRSP US Indices Database and Security Portfolio Assignment Module.

Published S&P 500 and NASDAQ Composite Index Data

The S&P 500 Composite Index is a value-weighted index created by Standard & Poor's. Since March 1957, the index contains 500 securities. Prior to that time, the index was called the S&P 90, containing 90 securities. These have been combined into a single time series. S&P Composite levels are collected from public sources such as the *Dow Jones News Service*, the *Wall Street Journal* and the *Standard & Poor's Statistical Service*.

The NASDAQ Composite Index is a value-weighted index created by the NASDAQ Stock Market.

Published S&P 500 and NASDAQ Composite Index Data are provided with the daily and monthly CRSPAccess Stock Files. Index levels and returns exclude dividends. As a result, the *Return with Dividends* variable returns a -88, or missing return code, for both indices. Total returns and membership data for the S&P 500 are available to subscribers of the CRSPAccess Index Files.

CRSP Stock File Capitalization Decile Indices

CRSP Stock File Capitalization Decile Indices are calculated for each of the Stock File Indices market groups. All securities excluding American Depositary Receipts on a given exchange or combination of exchanges are ranked according to capitalization and then divided into ten equal parts each rebalancing period.

The portfolios are rebalanced each year, using the security market capitalization at the end of the previous year to rank the securities. If a security starts trading in the middle of a year, its first capitalization of the year is used in the ranking. The largest securities are placed in portfolio 10 and the smallest in portfolio 1. A security not assigned to a portfolio is not used in the index and has its Portfolio Assignment set to 0.

Value-Weighted Index Returns including all dividends are calculated on each of the ten portfolios. Index levels are calculated based on an initial value of 100.0 on December 29, 1972.

Each set of decile indices represents one Index Group of index results and one Portfolio Type of portfolio assignments and statistics. Ten Index Series are created for each Portfolio Type.

CRSP Stock File Risk-Based Decile Indices

CRSP Stock File Risk-Based Decile Indices are created for the daily NYSE/AMEX and NASDAQ market combinations for two risk-based criteria. In these Market Segment Indices, portfolios are created by ranking securities according to a measurement of the risk of their returns. One ranking uses beta values computed using the methods

developed by Scholes and Williams (Myron Scholes and Joseph Williams, "Estimating Betas from Nonsynchronous Data", *Journal of Financial Economics*, vol 5, 1977, 309-327). The other ranking uses the annual standard deviation of the daily returns for its ranking.

The methodologies used to calculate these statistics are described in the CRSP Calculations section under Scholes-Williams Beta and Standard Deviation.

CRSP Stock File Risk-Based Decile Indices are rebalanced each year by ranking the statistics at the end of the previous year. If there are no data for the previous year for an issue but a valid statistic can be calculated for the current year, that statistic is used in the rankings. Portfolio 1 contains the securities with the highest statistics, and portfolio 10 contains the securities with the lowest statistics.

Once securities are assigned to portfolios, an equal-weighted total return index is calculated for each portfolio each calendar period. Trade-only security total returns are used for the NYSE/AMEX Beta Portfolios only. Index levels are calculated based on an initial value of 100.0 on December 29, 1972.

Each set of decile indices represents one Index Group of index results and one Portfolio Type of portfolio assignments and statistics. Ten Index Series are created for each Portfolio Type.

3.2 CRSP Cap-Based Portfolios

CRSP Cap-Based Portfolio Indices data are a monthly series based on portfolios that are rebalanced quarterly. The methodology used to calculate the series differs from the CRSP Stock File Capitalization Decile Indices.

The universe includes all common stocks listed on the NYSE, AMEX, and NASDAQ Global and Global Select Markets (formerly NASDAQ National Market prior to July 2006). Eligible companies listed on the NYSE are ranked into equally populated deciles. The largest capitalizations in each decile serve as the breakpoints that are applied to various exchange groupings of the universe.

Decile results are created for three exchange groups:

- NYSE only.
- NYSE and AMEX. AMEX data are added beginning July 1962.
- NYSE, AMEX and NASDAQ Global and Global Select Markets. The NASDAQ Global and Global Select Markets were formerly the NASDAQ National Market, which was added beginning April 1982.

Individual decile portfolios are created for each exchange group, the largest being in decile 1 and the smallest in decile 10. In addition to each decile portfolio, returns are calculated for the following: CRSP 1-2, CRSP 3-5, CRSP 6-8, CRSP 9-10, CRSP 6-10 and CRSP 1-10.

Companies becoming eligible or ineligible during a quarter are handled with the following rules:

Securities added during a quarter are assigned to appropriate portfolios when two consecutive month-end prices are available.

When a security's last price is a month-end price, its month's return is included in the portfolios' quarterly return.

When the month-end price is missing, a replacement month-end value is derived from the delisting return including merger terms, regional exchanges, etc. If the derived replacement month-end price is not available, the last available daily price is used.

If an issue becomes ineligible for an index in the middle of a quarter but is still active, such as after an exchange change or because the issue is leaving the NASDAQ Global or Global Select Markets, the issue is considered held until the end of the month and then dropped.

Index Total Returns, Index Capital Appreciation, and Index Income Returns are calculated from a value-weighted portfolio of securities in the portfolio each period. Index Levels are calculated for each of these returns series based on an investment of one dollar on December 25, 1925.

Only monthly indices and portfolio assignments are calculated for the Cap-Based Portfolios. Each of the three sets of Cap-Based Indices represents one Index Group of index results and one Portfolio Type of portfolio assignments and statistics. Seventeen Index Series, one for each decile and each composite, are created for each Portfolio Type.

3.3 CRSP Indices for the S&P 500® Universe

CRSP Indices for the S&P 500® Universe, formerly the S&P 90®, are standard CRSP Market Indices derived from CRSP Stock Files but include only issues from the CRSP stock data that are in the S&P 500® universe.

The CRSP Indices for the S&P 500® series contain value- and equal-weighted returns with and without dividends for a market of stocks in the S&P 500® universe. Daily and monthly data beginning December 25, 1925 are provided. The published S&P 500® index and returns are also included for comparison. For a security to be included in the CRSP indices for the S&P 500 Universe, it must have a price at the end of the current period, a price at the end of the previous period, and it must be a member of the S&P 500 Universe at the end of the current period. See CRSP Market Indices for the variables calculated and the methodology used.

Prior to March, 1957, the index contains 90 issues. CRSP does not have data for two securities that were part of the S&P 90® at different times between 1925 and 1931 as follows.

Company Name	Start Date	End Date
INT'L MERCANTILE MARINE PFD	31-dec-1925	22-jul-1929
STANDARD POWER & LIGHT "B"	06-feb-1930	16-nov-1931

Due to differences in handling mergers, reorganizations, and other major corporate actions, CRSP data and the S&P 500® universe do not always have a one-to-one mapping. In some cases this results in a short period where CRSP is missing prices or has multiple prices per company listed by S&P.

The Count of Securities Used is not always 500 (90 prior to March 1957) due to missing prices. Known reasons for missing prices are when-issued trading, halts, and suspensions.

CRSP Portfolios for the S&P 500 Universe

The CRSP Portfolios for the S&P 500 Universe include an alternate value- and equal-weighted version of the CRSP indexes for the S&P 500 Universe. The methodology differences are:

- Issues are selected based on membership in the S&P 500 at the end of the previous period instead of the end of the current period.
- Delisting returns are used to evaluate the value of securities that delist before the end of a period they were selected.

3.4 CRSP Treasury and Inflation Indices

The CRSP US Treasury and Inflation Series (CTI) Files are provided on a monthly frequency. The series contains returns adapted from the CRSP US Treasury Fixed Term Index Series, the CRSP Risk Free Rates File, and the US Government Consumer Price Index. These derived files offer 10 groups of indices: 30 year, 20 year, 10 year, 7 year, 5 year, 2 year, 1 year, 90 day, and 30 day target maturity indices, as well as the Consumer Price Index.

For fixed-term series with maturities of one year or greater, a representative Treasury bond or note for each series is selected. Available issues are filtered on the basis of their characteristics. Each month, the most recent non-callable, non-flower, and fully taxable issue closest to the target maturity is selected. If none are found, a second pass allows flower bonds. Note that all these series begin in 1941 or 1942 due to the lack of suitable issues in the early history.

For thirty and ninety day risk-free series, a representative Treasury bill for each series is selected. Each month the issue maturing closest to the target duration, as measured from the end of the previous month, is selected. Bills must have at least thirty days to their maturity date to be selected for the thirty day series. However, for ninety day series, bills with less than ninety days to maturity may be selected. Due to the lack of data, the selection process in periods prior to 1942 is somewhat subjective and the maturities of the selected issues may deviate more than several days from the thirty and ninety day targets. Where bills were not available, certificates or notes may have been used.

Exclusions may include:

- suspicious quotes,
- issues that did not mature on their next coupon payment data, or
- bid quotations that implied negative yields.

Each monthly return is calculated as price change plus interest, divided by last month's price. The returns and corresponding index values are set to –99 for months in which a return cannot be calculated, i.e. if the price is missing for either this month or last month, or if no valid issue was available.

The issue chosen for the 30, 20, 10, 7, 5, 2, and 1 year Fixed Term Index series for a given date was selected based on its length to maturity as of the date. The returns contained in these series are calculated under the assumption that the relevant issue is bought one month prior to the quote date and sold on the date.

The issue chosen for the 90 and 30 day Treasury Bill series on a given date was selected based on its length to maturity as of the month immediately prior to the date. The 90 and 30 day series returns were calculated on the basis of buying the relevant issue one month prior to the date and selling it on the date. For example, a 90 day bill return is calculated between a date approximately 90 days prior to the bill's maturity, and the date which is a month after this date. Likewise, a 30 day bill return is calculated between a date approximately 30 days prior to the bill's maturity, and the date which is a date one month later. In cases where the date chronologically approached or exceeded the maturity date, thereby making a final price unavailable, the return was calculated based on a final price of $100.

The associated index levels of the CRSP US Treasury and Inflation Series all have been initialized so that December 29, 1972 (19721229) equals 100. This facilitates comparison between the CTI Indices and Stock File Indices.

3.5 CRSP Select File Specifications[1]

Long Term Bond Selection

1. Select the 20-year bond that is the closest to having a term of at least 19.5 years to maturity at the beginning of the year. If more than one exists, choose the bond with the most current dated date (i.e. most recently issued).
2. If a 20-year bond does not meet the above criteria, choose the 25-year bond with at least 19.7 years to maturity at the beginning of the year. If more than one exists, choose the bond closest to 20 years to maturity.
3. If a 25-year bond does not meet the above criteria, choose the 30-year bond with at least 19.7 years to maturity at the beginning of the year. If more than one exists, choose the bond closest to 20 years to maturity on the quote date.

The bond chosen under any of the categories above cannot be dated any later than December 1st of the previous year for which the bond is being considered for inclusion in the index (i.e. dated date + one month < = quote date). Before 1942, only partially tax-exempt bonds (itax=2) are chosen because of the limited number of fully taxable bond issues. After 1942, only fully taxable issues are chosen (itax=1).

The bond is held for one full year in the index. Bonds chosen for this index are either non-callable or callable Treasury bonds with a type of Bond or Callable Bond. A 20-year bond can be selected from a universe of bonds that were issued as having a term to maturity of 7305-7693 days, a 25-year bond from an issue of 8766-9892 days, and a 30-year bond from an issue of 10955-11288 days.

Intermediate Term Bond Selection

1. Select the most currently issued 5 year bond with at least 5 years to maturity at the beginning of the calendar year.

[1]The CRSP Select Index Series is provided in ASCII only.

2. If a 5-year bond does not meet the above criteria select the next shortest maturity that is closest to 5 years to maturity on the quote date. For example, if a 7-year bond exists, choose the 7-year closest to 5 years to maturity. If a 7-year bond does not exist move up to the next highest maturity and so forth.

For the period 1934-1942, always choose a non-flower bond (iflwr of 1) and preference is given to a bond that is partially tax-exempt (itax of 2). If a partially tax-exempt bond does not meet the above criteria, choose a wholly tax-exempt bond (itax of 3). After 1942 only fully taxable non-flower bonds are chosen.

Callable and non-callable U.S. Treasury bonds and notes are considered for index inclusion. The issues are chosen from a universe of bonds issued with a term to maturity between 1000 to 7000 days to maturity.

Short Term Bond Selection

Choose the Treasury Bill closest to 90 days to maturity on the quote date. A bill can be within 4 days of target maturity, i.e. 90 days plus or minus 4 days. If a bill is not available use a certificate or a note.

3.6 CRSP Index Series and Groups

The following table contains a list of all CRSP Index Series by INDNO. For Index Groups, see page 37.

INDNO	Index Name	Daily Setid	Monthly Setid	Product
1000000	CRSP NYSE Value-Weighted Market Index	460	420	IF
1000001	CRSP NYSE Equal-Weighted Market Index	460	420	IF
1000002	CRSP NYSE Market Capitalization Decile 1	460	420	IF
1000003	CRSP NYSE Market Capitalization Decile 2	460	420	IF
1000004	CRSP NYSE Market Capitalization Decile 3	460	420	IF
1000005	CRSP NYSE Market Capitalization Decile 4	460	420	IF
1000006	CRSP NYSE Market Capitalization Decile 5	460	420	IF
1000007	CRSP NYSE Market Capitalization Decile 6	460	420	IF
1000008	CRSP NYSE Market Capitalization Decile 7	460	420	IF
1000009	CRSP NYSE Market Capitalization Decile 8	460	420	IF
1000010	CRSP NYSE Market Capitalization Decile 9	460	420	IF
1000011	CRSP NYSE Market Capitalization Decile 10	460	420	IF
1000012	CRSP NYSE Market Capitalization Deciles	440	400	IF
1000020	CRSP AMEX Value-Weighted Market Index	460	420	IF
1000021	CRSP AMEX Equal-Weighted Market Index	460	420	IF
1000022	CRSP AMEX Market Capitalization Decile 1	460	420	IF
1000023	CRSP AMEX Market Capitalization Decile 2	460	420	IF
1000024	CRSP AMEX Market Capitalization Decile 3	460	420	IF
1000025	CRSP AMEX Market Capitalization Decile 4	460	420	IF
1000026	CRSP AMEX Market Capitalization Decile 5	460	420	IF
1000027	CRSP AMEX Market Capitalization Decile 6	460	420	IF
1000028	CRSP AMEX Market Capitalization Decile 7	460	420	IF
1000029	CRSP AMEX Market Capitalization Decile 8	460	420	IF
1000030	CRSP AMEX Market Capitalization Decile 9	460	420	IF
1000031	CRSP AMEX Market Capitalization Decile 10	460	420	IF
1000032	CRSP AMEX Market Capitalization Deciles	440	400	IF
1000040	CRSP NYSE/AMEX Value-Weighted Market Index	460	420	IF
1000041	CRSP NYSE/AMEX Equal-Weighted Market Index	460	420	IF
1000042	CRSP NYSE/AMEX Market Capitalization Decile 1	460	420	IF
1000043	CRSP NYSE/AMEX Market Capitalization Decile 2	460	420	IF
1000044	CRSP NYSE/AMEX Market Capitalization Decile 3	460	420	IF
1000045	CRSP NYSE/AMEX Market Capitalization Decile 4	460	420	IF
1000046	CRSP NYSE/AMEX Market Capitalization Decile 5	460	420	IF
1000047	CRSP NYSE/AMEX Market Capitalization Decile 6	460	420	IF
1000048	CRSP NYSE/AMEX Market Capitalization Decile 7	460	420	IF
1000049	CRSP NYSE/AMEX Market Capitalization Decile 8	460	420	IF
1000050	CRSP NYSE/AMEX Market Capitalization Decile 9	460	420	IF
1000051	CRSP NYSE/AMEX Market Capitalization Decile 10	460	420	IF
1000052	CRSP NYSE/AMEX Market Capitalization Deciles	440	400	IF
1000053	CRSP NYSE/AMEX Trade-Only Value-Weighted Market Index	460	-	IF
1000060	CRSP NASDAQ Value-Weighted Market Index	460	420	IF
1000061	CRSP NASDAQ Equal-Weighted Market Index	460	420	IF
1000062	CRSP NASDAQ Market Capitalization Decile 1	460	420	IF
1000063	CRSP NASDAQ Market Capitalization Decile 2	460	420	IF
1000064	CRSP NASDAQ Market Capitalization Decile 3	460	420	IF
1000065	CRSP NASDAQ Market Capitalization Decile 4	460	420	IF
1000066	CRSP NASDAQ Market Capitalization Decile 5	460	420	IF
1000067	CRSP NASDAQ Market Capitalization Decile 6	460	420	IF
1000068	CRSP NASDAQ Market Capitalization Decile 7	460	420	IF
1000069	CRSP NASDAQ Market Capitalization Decile 8	460	420	IF
1000070	CRSP NASDAQ Market Capitalization Decile 9	460	420	IF
1000071	CRSP NASDAQ Market Capitalization Decile 10	460	420	IF

INDNO	Index Name	Daily Setid	Monthly Setid	Product
1000072	CRSP NASDAQ Market Capitalization Deciles	440	400	IF
1000080	CRSP NYSE/AMEX/NASDAQ Value-Weighted Market Index	460	420	STK, IF
1000081	CRSP NYSE/AMEX/NASDAQ Equal-Weighted Market Index	460	420	STK, IF
1000082	CRSP NYSE/AMEX/NASDAQ Market Capitalization Decile 1	460	420	IF
1000083	CRSP NYSE/AMEX/NASDAQ Market Capitalization Decile 2	460	420	IF
1000084	CRSP NYSE/AMEX/NASDAQ Market Capitalization Decile 3	460	420	IF
1000085	CRSP NYSE/AMEX/NASDAQ Market Capitalization Decile 4	460	420	IF
1000086	CRSP NYSE/AMEX/NASDAQ Market Capitalization Decile 5	460	420	IF
1000087	CRSP NYSE/AMEX/NASDAQ Market Capitalization Decile 6	460	420	IF
1000088	CRSP NYSE/AMEX/NASDAQ Market Capitalization Decile 7	460	420	IF
1000089	CRSP NYSE/AMEX/NASDAQ Market Capitalization Decile 8	460	420	IF
1000090	CRSP NYSE/AMEX/NASDAQ Market Capitalization Decile 9	460	420	IF
1000091	CRSP NYSE/AMEX/NASDAQ Market Capitalization Decile 10	460	420	IF
1000092	CRSP NYSE/AMEX/NASDAQ Market Capitalization Deciles	440	400	IF
1000102	CRSP NYSE/AMEX Beta Decile 1	460	-	IF
1000103	CRSP NYSE/AMEX Beta Decile 2	460	-	IF
1000104	CRSP NYSE/AMEX Beta Decile 3	460	-	IF
1000105	CRSP NYSE/AMEX Beta Decile 4	460	-	IF
1000106	CRSP NYSE/AMEX Beta Decile 5	460	-	IF
1000107	CRSP NYSE/AMEX Beta Decile 6	460	-	IF
1000108	CRSP NYSE/AMEX Beta Decile 7	460	-	IF
1000109	CRSP NYSE/AMEX Beta Decile 8	460	-	IF
1000110	CRSP NYSE/AMEX Beta Decile 9	460	-	IF
1000111	CRSP NYSE/AMEX Beta Decile 10	460	-	IF
1000112	CRSP NYSE/AMEX Beta Deciles	440	-	IF
1000122	CRSP NYSE/AMEX Standard Deviation Decile 1	460	-	IF
1000123	CRSP NYSE/AMEX Standard Deviation Decile 2	460	-	IF
1000124	CRSP NYSE/AMEX Standard Deviation Decile 3	460	-	IF
1000125	CRSP NYSE/AMEX Standard Deviation Decile 4	460	-	IF
1000126	CRSP NYSE/AMEX Standard Deviation Decile 5	460	-	IF
1000127	CRSP NYSE/AMEX Standard Deviation Decile 6	460	-	IF
1000128	CRSP NYSE/AMEX Standard Deviation Decile 7	460	-	IF
1000129	CRSP NYSE/AMEX Standard Deviation Decile 8	460	-	IF
1000130	CRSP NYSE/AMEX Standard Deviation Decile 9	460	-	IF
1000131	CRSP NYSE/AMEX Standard Deviation Decile 10	460	-	IF
1000132	CRSP NYSE/AMEX Standard Deviation Deciles	440	-	IF
1000142	CRSP NASDAQ Beta Decile 1	460	-	IF
1000143	CRSP NASDAQ Beta Decile 2	460	-	IF
1000144	CRSP NASDAQ Beta Decile 3	460	-	IF
1000145	CRSP NASDAQ Beta Decile 4	460	-	IF
1000146	CRSP NASDAQ Beta Decile 5	460	-	IF
1000147	CRSP NASDAQ Beta Decile 6	460	-	IF
1000148	CRSP NASDAQ Beta Decile 7	460	-	IF
1000149	CRSP NASDAQ Beta Decile 8	460	-	IF
1000150	CRSP NASDAQ Beta Decile 9	460	-	IF
1000151	CRSP NASDAQ Beta Decile 10	460	-	IF
1000152	CRSP NASDAQ Beta Deciles	440	-	IF
1000162	CRSP NASDAQ Standard Deviation Decile 1	460	-	IF
1000163	CRSP NASDAQ Standard Deviation Decile 2	460	-	IF
1000164	CRSP NASDAQ Standard Deviation Decile 3	460	-	IF
1000165	CRSP NASDAQ Standard Deviation Decile 4	460	-	IF
1000166	CRSP NASDAQ Standard Deviation Decile 5	460	-	IF
1000167	CRSP NASDAQ Standard Deviation Decile 6	460	-	IF
1000168	CRSP NASDAQ Standard Deviation Decile 7	460	-	IF
1000169	CRSP NASDAQ Standard Deviation Decile 8	460	-	IF
1000170	CRSP NASDAQ Standard Deviation Decile 9	460	-	IF

INDNO	Index Name	Daily Setid	Monthly Setid	Product
1000171	CRSP NASDAQ Standard Deviation Decile 10	460	-	IF
1000172	CRSP NASDAQ Standard Deviation Deciles	440	-	IF
1000180	CRSP Arca Value-Weighted Market Index	460	440	IF
1000181	CRSP Arca Equal-Weighted Market Index	460	440	IF
1000200	CRSP NYSE/AMEX/NASDAQ/Arca Value-Weighted Market Index	460	440	IF
1000201	CRSP NYSE/AMEX/NASDAQ/Arca Equal-Weighted Market Index	460	440	IF
1000300	CRSP NYSE Cap-Based Portfolio 1	-	420	IF
1000301	CRSP NYSE Cap-Based Portfolio 2	-	420	IF
1000302	CRSP NYSE Cap-Based Portfolio 3	-	420	IF
1000303	CRSP NYSE Cap-Based Portfolio 4	-	420	IF
1000304	CRSP NYSE Cap-Based Portfolio 5	-	420	IF
1000305	CRSP NYSE Cap-Based Portfolio 6	-	420	IF
1000306	CRSP NYSE Cap-Based Portfolio 7	-	420	IF
1000307	CRSP NYSE Cap-Based Portfolio 8	-	420	IF
1000308	CRSP NYSE Cap-Based Portfolio 9	-	420	IF
1000309	CRSP NYSE Cap-Based Portfolio 10	-	420	IF
1000310	CRSP NYSE Cap-Based Portfolio 1-2	-	420	IF
1000311	CRSP NYSE Cap-Based Portfolio 3-5	-	420	IF
1000312	CRSP NYSE Cap-Based Portfolio 6-8	-	420	IF
1000313	CRSP NYSE Cap-Based Portfolio 9-10	-	420	IF
1000314	CRSP NYSE Cap-Based Portfolio 1-5	-	420	IF
1000315	CRSP NYSE Cap-Based Portfolio 6-10	-	420	IF
1000316	CRSP NYSE Cap-Based Portfolio Market	-	420	IF
1000317	CRSP NYSE Cap-Based Portfolios	-	400	IF
1000320	CRSP NYSE/AMEX Cap-Based Portfolio 1	-	420	IF
1000321	CRSP NYSE/AMEX Cap-Based Portfolio 2	-	420	IF
1000322	CRSP NYSE/AMEX Cap-Based Portfolio 3	-	420	IF
1000323	CRSP NYSE/AMEX Cap-Based Portfolio 4	-	420	IF
1000324	CRSP NYSE/AMEX Cap-Based Portfolio 5	-	420	IF
1000325	CRSP NYSE/AMEX Cap-Based Portfolio 6	-	420	IF
1000326	CRSP NYSE/AMEX Cap-Based Portfolio 7	-	420	IF
1000327	CRSP NYSE/AMEX Cap-Based Portfolio 8	-	420	IF
1000328	CRSP NYSE/AMEX Cap-Based Portfolio 9	-	420	IF
1000329	CRSP NYSE/AMEX Cap-Based Portfolio 10	-	420	IF
1000330	CRSP NYSE/AMEX Cap-Based Portfolio 1-2	-	420	IF
1000331	CRSP NYSE/AMEX Cap-Based Portfolio 3-5	-	420	IF
1000332	CRSP NYSE/AMEX Cap-Based Portfolio 6-8	-	420	IF
1000333	CRSP NYSE/AMEX Cap-Based Portfolio 9-10	-	420	IF
1000334	CRSP NYSE/AMEX Cap-Based Portfolio 1-5	-	420	IF
1000335	CRSP NYSE/AMEX Cap-Based Portfolio 6-10	-	420	IF
1000336	CRSP NYSE/AMEX Cap-Based Portfolio Market	-	420	IF
1000337	CRSP NYSE/AMEX Cap-Based Portfolios	-	400	IF
1000340	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 1	-	420	IF
1000341	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 2	-	420	IF
1000342	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 3	-	420	IF
1000343	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 4	-	420	IF
1000344	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 5	-	420	IF
1000345	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 6	-	420	IF
1000346	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 7	-	420	IF
1000347	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 8	-	420	IF
1000348	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 9	-	420	IF
1000349	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 10	-	420	IF
1000350	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 1-2	-	420	IF
1000351	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 3-5	-	420	IF
1000352	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 6-8	-	420	IF
1000353	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 9-10	-	420	IF

INDNO	Index Name	Daily Setid	Monthly Setid	Product
1000354	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 1-5	-	420	IF
1000355	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio 6-10	-	420	IF
1000356	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolio Market	-	420	IF
1000357	CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolios	-	400	IF
1000500	CRSP Value-Weighted Index of the S&P 500 Universe	460	420	IF
1000501	CRSP Equal-Weighted Index of the S&P 500 Universe	460	420	IF
1000502	S&P 500 Composite	460	420	IF
1000503	NASDAQ Composite	460	420	IF
1000510	CRSP Value-Weighted Portfolios of the S&P 500 Universe	460	420	IF
1000511	CRSP Equal-Weighted Portfolios of the S&P 500 Universe	460	420	IF
1000700	CRSP 30-Year Bond Returns	-	420	IF
1000701	CRSP 20-Year Bond Returns	-	420	IF
1000702	CRSP 10-Year Bond Returns	-	420	IF
1000703	CRSP 7-Year Bond Returns	-	420	IF
1000704	CRSP 5-Year Bond Returns	-	420	IF
1000705	CRSP 2-Year Bond Returns	-	420	IF
1000706	CRSP 1-Year Bond Returns	-	420	IF
1000707	CRSP 90-Day Bill Returns	-	420	IF
1000708	CRSP 30-Day Bill Returns	-	420	IF
1000709	Consumer Price Index	-	420	IF

The following table contains a list of all CRSP Index Groups by INDNO.

CRSP Index Groups by INDNO

Index Groups	INDNO	Daily	Monthly	Product Availability
CRSP NYSE Market Capitalization Deciles	1000012	Yes	Yes	IF
CRSP AMEX Market Capitalization Deciles	1000032	Yes	Yes	IF
CRSP NYSE/AMEX Market Capitalization Deciles	1000052	Yes	Yes	IF
CRSP NASDAQ Market Capitalization Deciles	1000072	Yes	Yes	IF
CRSP NYSE/AMEX/NASDAQ Market Capitalization Deciles	1000092	Yes	Yes	IF
CRSP NYSE/AMEX Beta Deciles	1000112	Yes	-	IF
CRSP NYSE/AMEX Standard Deviation Deciles	1000132	Yes	-	IF
CRSP NASDAQ Beta Deciles	1000152	Yes	-	IF
CRSP NASDAQ Standard Deviation Deciles	1000172	Yes	-	IF
CRSP NYSE Cap-Based Portfolios	1000317	-	Yes	IF
CRSP NYSE/AMEX Cap-Based Portfolios	1000337	-	Yes	IF
CRSP NYSE/AMEX/NASDAQ National Market Cap-Based Portfolios	1000357	-	Yes	IF

3.7 Portfolio Types Defined by CRSP

Portfolio Types Defined by CRSP

Portfolio Type Description	Rebalancing Calendar	INDNO	Daily Portfolio Type	Monthly Portfolio Type	Product Availability
NYSE/AMEX/NASDAQ Capitalization Deciles	Annual	1000092	1	1	DA, D6, MA, M6, IF, DI, MI
NYSE/AMEX Capitalization Deciles	Annual	1000052	2	2	IF
NASDAQ Capitalization Deciles	Annual	1000072	3	3	IF
NYSE Capitalization Deciles	Annual	1000012	4	4	IF
AMEX Capitalization Deciles	Annual	1000032	5	5	IF
NYSE/AMEX Beta Deciles	Annual	1000112	6	-	IF
NYSE/AMEX Standard Deviation Deciles	Annual	1000132	7	-	IF
NASDAQ Beta Deciles	Annual	1000152	8	-	IF
NASDAQ Standard Deviation Deciles	Annual	1000172	9	-	IF
Cap-Based NYSE/AMEX/NASDAQ National Market Portfolios	Quarterly	1000357	-	6	IF
Cap-Based NYSE Portfolios	Quarterly	1000317	-	7	IF
Cap-Based NYSE/AMEX Portfolios	Quarterly	1000337	-	8	IF

CHAPTER 4: DATA DEFINITIONS

This section contains the CRSP stock and indices variables organized alphabetically by name. Each variable includes a detailed definition and a usage table. The usage table contains the following items:

Primary Concept(s)		Always contains the name of the primary data groups described in the data structure section, and may contain references to calculations or index methodologies.					Data Type	Type of data; character, integer number, real number, or structure.
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	The first date daily data is available for the data item. 1925 begins 19251231.	Monthly:	The first date monthly data is available for the data item. 1925 begins 19251231.	Unit	The unit of the data items. (See the table below this section for possible values.)	Database Format	CRSPAccess
C Usage				FORTRAN-95 UsageThese fields contain shortcuts where they are available. See the Programmer's Guide for complete usage information.			Product Type(s)	Data file needed to access the data variable. Possible values are: `STK`-Stock data, `STK*`-Stock data, limited index sample included with stock calendar/indices data. More functionality available with full Index product, `STK‡`-Stock data item requires index product to utilize, `IND`-Indices Data, and `IND‡`- Index data item requires stock data to utilize.
Object	The name of the CRSPAccess data object needed to access the variable using C.			Type or Subtype	The name of the FORTRAN-95 Data Type, Subtype used to access the variable.		*ts_print* Daily Usage	The data item/`subno` (required combination) used to access the variable from a daily CRSPAccess database with the *ts_print* utility. See the Utilities Guide for usage of *ts_print*.
Array	The name of the CRSPAccess array used to access the variable using C.			Member and/or Arrays	The name of the FORTRAN-95 array used to access the variable.		*ts_print* Monthly Usage	Same as daily, but for use with a monthly CRSPAccess database.
Element	The mnemonic name of the structure element or structure within the array used to access the variable using C.			Element	The name of the FORTRAN-95 element used to access the variable.		*stk_print* or *ind_print* Option(s)	Options that can be used to access the variable from the *stk_print* utility. See the Utilities Guide for usage of *stk_print*.

See Chapter 2 for an overview of the databases and a summary of the organization of the variables; the *Programmers Guide* for C and FORTRAN programming usage of a CRSPAccess database; the *FORTRAN-77 Migration Guide* for FORTRAN-77 programming information.

The *ts_print* and *stk_print* items included in this section are presently direct mappings and do not include derived or otherwise modified data items. For utility program usage and available data items, see the Utilities Guide.

Variables names are italicized for easy recognition and cross reference.

Tables containing variables based on programming access are included in the Database Structure Section of the Data Description Guide and the FORTRAN-77 Migration Guide to cross reference the names and mnemonics of the variables and arrays in the Stock and Indices data. These tables are sorted in array name order.

Unit Values Include:

Value	Value	Value
% Rate	YYYYYMMDD date	YYYYMM date
Array Index	Code	Conversion Factor
Count	(Dependant on data loaded)	description
Id	Mathematical value	Meta data
Pointer	range	Ratio
Set (*)	USD	USD in thousands

Acquiring PERMCO

Acquiring PERMCO is the PERMCO of another company linked to a distribution. If the *Acquiring PERMNO* is non-zero and represents an associated security, *Acquiring PERMCO* is set to the PERMCO of that security. If *Acquiring PERMNO* is less than 1000, then *Acquiring PERMCO* can still be set. In this case, it represents a link to a company tracked by CRSP rather than a specific issue. For example, if a company pays cash to shareholders in a merger, then the *Acquiring PERMCO* is set to the *PERMCO* of that company.

Acquiring PERMCO is zero if not applicable, unknown, or associated with a company not tracked by CRSP. Data in this field is incomplete prior to 1985.

Primary Concept(s)	Distribution Event Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`dists_arr`			Type or Subtype	`dists_arr`		*ts_print* Daily Usage	n/a
Array	`dists[]`			Member and/or Array	`dists()`		*ts_print* Monthly Usage	n/a
Element	`accomp`			Element	`accomp`		*stk_print* Option(s)	`/di`

Acquiring PERMNO

Acquiring PERMNO is the PERMNO of another security linked to a distribution where a stock was received in a spin-off, exchange, merger, or other distribution event. It can also link to a security that was acquired in a merger causing a shares increase.

Acquiring PERMNO is set to a number less than 1000 if inapplicable or unknown. If multiple distributions exist with the same *Distribution Code* and *Ex-Distribution Date*, they are numbered in the *Acquiring PERMNO* field. Data in this field are incomplete prior to 1985. *Acquiring PERMNO* may point to four-digit securities. Data for these securities are not included in the databases. *Acquiring PERMNO* values between 1 and 9 do not represent securities. These values as used to distinguish multiple distribution records with the same *Distribution Code* on the same *Ex-Distribution Date*.

Primary Concept(s)	Distribution Event Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`dists_arr`			Type or Subtype	`dists_arr`		*ts_print* Daily Usage	n/a
Array	`dists[]`			Member and/or Array	`dists()`		*ts_print* Monthly Usage	n/a
Element	`acperm`			Element	`acperm`		*stk_print* Option(s)	`/di`

Amount After Delisting

Amount After Delisting is the value of a security after it delists from an exchange. The amount can be either an off-exchange price, an off-exchange price quote, or the sum of a series of distribution payments. The *Amount After Delisting* is used to calculate the *Delisting Return*. This amount is set to zero if the security is still active, if no price or payment information is available, or if the stock is worthless.

Monthly: If no value after the *Delisting Date* exists, but daily prices exist after the previous month's last trading date, then the *Amount After Delisting* is set to the last daily trading value found in the *Price or Bid/Ask Average*. This price is a daily price for activity that occurred during the delisting month.

Primary Concept(s)	Delisting History Array						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	USD	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`delist_arr`			Type or Subtype	`delist_arr`		*ts_print* Daily Usage	n/a
Array	`delist[]`			Member and/or Array	`delist()`		*ts_print* Monthly Usage	n/a
Element	`dlamt`			Element	`dlamt`		*stk_print* Option(s)	`/de`

Array Structure Size

Array Structure Size is the number of bytes needed in each structure element for this array type in a CRSPAccess object structure.

Primary Concept(s)	Base CRSPAccess Data Structure(s); Time Series Objects, Event Array Objects, Header Objects						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	-	Monthly:	-	Unit	Set (no. of bytes)	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_*`			Type or Subtype			*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays			*ts_print* Monthly Usage	n/a
Element	`size_of_array_width`			Element			*stk_print* or *ind_print* Option(s)	n/a

Array Type Code

Array Type Code is an integer code which defines the type of data in a CRSPAccess object structure array. It can define a basic data type or a CRSP-defined structure.

Primary Concept(s)	Base CRSPAccess Data Structure(s); Time Series Objects, Event Array Objects, Header Objects						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	-	Monthly:	-	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_*`			Type or Subtype	`crsp_ts` or `crsp_array`		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`arrtype`			Element	`arrtype`		*stk_print* or *ind_print* Option(s)	n/a

Ask

Ask is available both daily and monthly for all securities on the NYSE, AMEX, NASDAQ, and NYSE Arca. Ask prices for NASDAQ are handled a little differently than for NYSE/AMEX and outlined as follows:

NASDAQ

Ask is available for issues trading on the NASDAQ Stock Market during time periods when Ask or High Price can contain the high price.

Since July 1980, NASDAQ has used the inside quotation as the closing bid and ask, with the close being at 4:00PM Eastern time. The inside quotation is the highest bid and lowest ask for each trade date.

Ask is reported for all NASDAQ National Market securities since November1, 1982 and for all NASDAQ securities since June 15, 1992, with the following exceptions due to source limitations:

- Ask is missing for 15 NASDAQ National Market securities in December 1982.
- Ask is missing for all NASDAQ National Market securities in February 1986.

NYSE/ AMEX

The ask for NYSE and AMEX securities is not the inside quotation, but the ask price from the last representative quote before the markets close for each trading date. Due to source limitations, only an unrepresentative quote was available on many days. This unrepresentative quote showed very large spreads, frequently a bid of a penny and an ask of approximately double the price. These were usually posted by a market marker not on the primary listed exchange, who was required to post a quote but not interested in making a trade. From 1992 on, Bid and Ask were set to 0 when CRSP determined that the available quote was unrepresentative of trading activity, pending further research.

Ask data for NYSE are available from December 31, 1925 through the most currently completed month for securities when no closing price is available. Between December 31, 1925 and February 23, 1942, a continuous series of ask data are available whether or not a closing price is available. Between February 24, 1942 and December 27, 1992, ask is available only in cases when a closing price is missing. Beginning December 28, 1992, a continuous series of ask data are available.

Primary Concept(s)	Supplemental NASDAQ Time Series						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	USD	Database Format(s)	`CRSPAccess` - daily only
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`ask_ts`		Type or Subtype	`ask_ts`			*ts_print* Daily Usage	`ask/0`
Array	`ask[]`		Member and/or Array	`ask()`			*ts_print* Monthly Usage	`mask/0`
Element	n/a		Element	n/a			*stk_print* Option(s)	`/pa`

Ask or High Price

Daily: *Ask or High Price* is the highest trading price during the day, or the closing ask price on days when the closing price is not available. The field is set to zero if no *Ask or High Price* is available. If the Price or Bid/Ask Average is negative, this field contains the closing ask. If positive, this field contains the highest trade.

Daily trading prices for the NASDAQ National Market securities were first reported November 1, 1982. Daily trading prices for The NASDAQ SmallCap Market were first reported June 15, 1992. Therefore, *Ask or High Price* for NASDAQ securities is always an ask before these dates.

Monthly: Monthly files contain the highest daily *Price or Bid/Ask Average* during the month. The field is set to zero when no *Price or Bid/Ask Average* was available during the month a value is found for incomplete months. If *Price or Bid/Ask Average* contains any bid/ask averages, these will be marked with a negative symbol. The absolute value of *Price or Bid/Ask Average* is used to select the highest, but the sign is preserved if a bid/ask average is selected.

Primary Concept(s)	Price, Volume, and Return Time Series Arrays						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	USD	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`askhi_ts`			Type or Subtype	`askhi_ts`		*ts_print* Daily Usage	`askhi/0`
Array	`askhi[]`			Member and/or Array	`askhi()`		*ts_print* Monthly Usage	`maskhi/0`
Element	n/a			Element	n/a		*stk_print* Option(s)	`/ph, /dd`

Begin of Group Data

Begin of Group Data is the first date the group data is valid in YYYYMMDD format.

Primary Concept(s)	Group Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`group_arr[i], i+1` is `grouptype`			Type or Subtype	`group_arr(i) i` is `grouptype`		*ts_print* Daily Usage	n/a
Array	`group[i,]`			Member and/or Arrays	`group()`		*ts_print* Monthly Usage	n/a
Element	`grpdt`			Element	`grpdt`		*stk_print* Option(s)	`/g#, #` is `grouptype`

Begin of Stock Data

Begin of Stock Data is the date that data begins for the security, in `YYYYMMDD` format. It is the date of the first period in the time series arrays and is always greater than zero.

Primary Concept(s)	Header Identification and Summary Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	n/a
Array	`header`			Member and/or Arrays	`n/a`		*ts_print* Monthly Usage	n/a
Element	`begdt`			Element	`begdt`		*stk_print* Option(s)	`/hh`, `/hr`, `/hrl`, `/hn`

Begin of Valid Data

Begin of Valid Data is the index of the first calendar period with valid data in a time series. If no data of this type are available, it is set to zero. The *Calendar Trading Date* at this index is the date of the first calendar period with data of this type.

Primary Concept(s)	Base CRSPAccess Data Structure(s); Time Series Objects						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Array index	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_TIMESERIES`			Type or Subtype	`crsp_ts`		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`beg`			Element	`beg`		*stk_print* or *ind_print* Option(s)	n/a

Bid

Bid is available both daily and monthly for all securities on the NYSE, AMEX, NASDAQ, and NYSE Arca exchanges. Bid prices for NASDAQ are handled a little differently than for NYSE/AMEX and outlined as follows:

NASDAQ

Bid is available for issues trading on the NASDAQ Stock Market during time periods when Bid or Low Price can contain the high price.

Since July 1980, NASDAQ has used the inside quotation as the closing bid and ask, with the close being at 4:00PM Eastern time. The inside quotation is the highest bid and lowest ask for each trade date.

Bid is reported for all NASDAQ National Market securities since November1, 1982 and for all NASDAQ securities since June 15, 1992, with the following exceptions due to source limitations:

- Bid is missing for 15 NASDAQ National Market securities in December 1982.
- Bid is missing for all NASDAQ National Market securities in February 1986.

NYSE/ AMEX

The bid for NYSE and AMEX securities is not the inside quotation, but the bid price from the last representative quote before the markets close for each trading date. Due to source limitations, only an unrepresentative quote was available on many days. This unrepresentative quote showed very large spreads, frequently a bid of a penny and an ask of approximately double the price. These were usually posted by a market marker not on the primary listed exchange, who was required to post a quote but not interested in making a trade. From 1992 on, Bid and Ask are set to 0 when CRSP determined that the available quote was unrepresentative of trading activity, pending further research.

Bid data for NYSE are available from December 31, 1925 through the most currently completed month for securities when no closing price is available. Between December 31, 1925 and February 23, 1942, a continuous series of bid data are available whether or not a closing price is available. Between February 24, 1942 and December 27, 1992, bid is available only in cases when a closing price is missing. Beginning December 28, 1992, a continuous series of bid data are available.

Primary Concept(s)	Supplemental NASDAQ Time Series						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	USD	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`bid_ts`			Type or Subtype	`bid_ts`		*ts_print* Daily Usage	`bid/0`
Array	`bid[]`			Member and/or Array	`bid()`		*ts_print* Monthly Usage	`mbid/0`
Element	n/a			Element	n/a		*stk_print* Option(s)	`/pb`

Bid or Low Price

Daily: *Bid or Low Price* is the lowest trading price during the day or the closing bid price on days when the closing price is not available. The field is set to zero if no *Bid or Low Price* is available.

Daily trading prices for the NASDAQ National Market securities were first reported November 1, 1982. Daily trading prices for The NASDAQ SmallCap Market were first reported June 15, 1992. Therefore, *Bid or Low Price* for NASDAQ securities is always a bid before these dates.

Monthly: Monthly files contain the lowest daily *Price or Bid/Ask Average* during the month. Closing price values are positive, bid/ask averages negative. The negative sign is a symbol used to differentiate between price and bid/ask average. The bid/ask average does not have a negative value, and the sign is ignored for ranking. The field is set to zero when no ask or bid/ask average is available.

Primary Concept(s)	Price, Volume, and Return Time Series Arrays						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	USD	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`bidlo_ts`			Type or Subtype	`bidlo_ts`		*ts_print* Daily Usage	`bidlo/0`
Array	`bidlo[]`			Member and/or Array	`bidlo()`		*ts_print* Monthly Usage	`mbidlo/0`
Element	n/a			Element	n/a		*stk_print* Option(s)	`/pl`, `/dd`

Calendar Associated with a Time Series

Calendar Associated with a Time Series is a pointer in a CRSP time series object to the associated CRSPAccess calendar structure needed to assign the time to data in a time series array.

Primary Concept(s)	Base CRSPAccess Data Structure(s); Calendars, Time Series Objects						Data Type	structure
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Array index	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_TIMESERIES`			Type or Subtype	`crsp_ts`		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`cal`			Element	`cal`		*stk_print* or *ind_print* Option(s)	n/a

Calendar Identification Number

Calendar Identification Number is an integer code assigned by CRSP to trading calendars. A *Calendar Name* and *Calendar Identification Number* identify each calendar. The calendars supported in CRSPAccess databases are:

Calendars	Calendar Identification Number	Calendar Name	Beginning Date
Daily	100	Daily Trading Calendar	19251231
Monthly	101	Month-End Trading Calendar	19251231
Annual	300	Annual Trading Calendar	19251231
Quarterly	310	Quarterly Trading Calendar	19251231
Weekly	500	Weekly Trading Calendar	19260102

Primary Concept(s)	Base CRSPAccess Data Structure(s); Calendar Objects						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_CAL`			Type or Subtype	`crsp_cal`		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`calid`			Element	`calid`		*stk_print* or *ind_print* Option(s)	n/a

Calendar Identification Number of Assignment Calendar

Calendar Identification Number of Assignment Calendar identifies a calendar that determines the dates when index breakpoints and buy/sell rules are valid. The assignment calendar is used when using rebalancing information to assign issues to a portfolio. The calendar periods of the *Calendar Identification Number of Rebalancing Calendar*, *Calendar Identification Number of Assignment Calendar*, and *Calendar Identification Number of Calculations Calendar* are synchronized, although the actual date ranges for each period may differ. The assignment calendar uses the same calendars listed in *Calendar Identification Number*.

Primary Concept(s)	Index Header, Calendars						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`assign`		*ts_print* Monthly Usage	n/a
Element	`assign.assigncal`			Element	`assigncal`		*ind_print* Option(s)	`/hr`

Calendar Identification Number of Calculations Calendar

Calendar Identification Number of Calculations Calendar identifies a calendar that determines the range of dates used to calculate statistics to form portfolios. The calendar periods of the *Calendar Identification Number of Rebalancing Calendar*, *Calendar Identification Number of Assignment Calendar*, and *Calendar Identification Number of Calculations Calendar* are synchronized, although the actual date ranges for each period number may differ. The calculations calendar uses the same calendars described in *Calendar Identification Number*.

Primary Concept(s)	Index Header, Calendars						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`assign`		*ts_print* Monthly Usage	n/a
Element	`assign.calccal`			Element	`calccal`		*ind_print* Option(s)	`/hr`

Calendar Identification Number of Rebalancing Calendar

Calendar Identification Number of Rebalancing Calendar identifies a calendar that determines the time periods when the portfolios in the index are held. The new portfolio universe is held from the end of one period in the rebalancing calendar until the end of the next period. The calendar period numbers of the *Calendar Identification Number of Rebalancing Calendar*, *Calendar Identification Number of Assignment Calendar*, and the *Calendar Identification Number of Calculations Calendar* are synchronized, although the actual date ranges for each period number may differ. The rebalancing calendar uses the same calendars listed in *Calendar Identification Number*.

Primary Concept(s)	Index Header, **calendars**						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`assign`		*ts_print* Monthly Usage	n/a
Element	`assign.rebalcal`			Element	`rebalcal`		*ind_print* Option(s)	`/hr`

Calendar Name

Calendar Name is a text description of a calendar. These include: Daily Trading Calendar, Month-End Trading Calendar, Annual Trading Calendar, Quarterly Trading Calendar, Weekly Trading Calendar.

Primary Concept(s)	Base CRSPAccess Data Structure(s); Calendar Objects						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1926	Monthly:	1925	Unit	Description	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_CAL`			Type or Subtype	`crsp_cal`		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`name`			Element	`name`		*stk_print* or *ind_print* Option(s)	n/a

Calendar Period Grouping Identifier

Calendar Period Grouping Identifiers are integers assigned for each trading period in a calendar. These identifiers can be used as alternate names to *Calendar Trading Date* for the calendar periods. The values stored for each period in *Calendar Period Grouping Identifier* for the current calendars are:

Calendar Name	Date Format
Daily Trading Calendar	`YYYYWW`
Month-End Trading Calendar	`YYYYMM`
Annual Trading Calendar	`YYYY`
Quarterly Trading Calendar	`YYYYMM`
Weekly Trading Calendar	`YYYYWW`

where `YYYY` is the 4-digit year, `MM` the month, and `WW` the week within the current year. The last week of the previous year is continued for the entire week when the year changes.

Primary Concept(s)	Base CRSPAccess Data Structure(s); Calendar Objects						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Description	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_CAL`			Type or Subtype	`crsp_cal`		*ts_print* Daily Usage	n/a
Array	`callist[]`			Member and/or Arrays	`callist()`		*ts_print* Monthly Usage	n/a
Element	n/a			Element	n/a		*stk_print* or *ind_print* Option(s)	n/a

Calendar Time Period Description Code

Calendar Time Period Description Code is a code that indicates the type of calendar array best used to read a CRSP Stock or Indices time series. This code equals 2, indicating that calendar time periods are identified by the last trading date in the period using *Calendar Trading Date*.

Primary Concept(s)	Base CRSPAccess Data Structure(s); Time Series Objects						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_TIMESERIES`			Type or Subtype	`crsp_ts`		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`caltype`			Element	`caltype`		*stk_print* or *ind_print* Option(s)	n/a

Calendar Trading Date

Calendar Trading Date contains the list of trading dates for a CRSP calendar. Each date represents the last date in a calendar period, in `YYYYMMDD` (year, month, day) format. These dates begin in the first element of the array and continue to the *Number of Periods in Calendar*. Calendar dates for weekends and trading holidays are not included.

Primary Concept(s)	Base CRSPAccess Data Structure(s);Calendar Objects						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_CAL`			Type or Subtype	`crsp_cal`		*ts_print* Daily Usage	`caldt/0`
Array	`caldt[]`			Member and/or Array	`caldt()`		*ts_print* Monthly Usage	`mcaldt/0`
Element	n/a			Element	n/a		*stk_print* or *ind_print* Option(s)	all time series use `/dt` for range restriction

Calendar Type Availability Flag

Calendar Type Availability Flag identifies the calendar available for use with the data. It is set to 1 if *Calendar Period Grouping Identifier* is available, 2 if *Calendar Trading Date* is available, and 3 if both are available.

Primary Concept(s)	Base CRSPAccess Data Structure(s); Calendar Objects						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_CAL`			Type or Subtype	`crsp_cal`		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`loadflag`			Element	`loadflag`		*stk_print* or *ind_print* Option(s)	n/a

Company Name

Company Name is the name of the company at the time of its name history record. CRSP allocates a 35 character name description field for all securities. Preference is given to the spellings and abbreviations provided in Standard & Poor's CUSIP Directory. In cases where name sources provide descriptions in excess of 35 characters, CRSP furnishes its own abbreviations.Daily: The daily file includes all historical names included in the file beginning in 1925.

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`comnam/0`
Array	`names`			Member and/or Array	`names`		*ts_print* Monthly Usage	`mcomnam/0`
Element	`comnam`			Element	`comnam`		*stk_print* Option(s)	/n, `/xn`[1]

[1] when used with `/fs`

Company Name - Header

Company Name - Header is the most current *Company Name* tracked in the specific security's name history array. Names can be up to 35 characters long.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage Id			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	`comnam|SUBNO2`
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	`mcomnam|SUBNO2`
Element	`hcomnam`			Element	`hcomnam`		*stk_print* Option(s)	/hh, `/hr`, `/hrl`, `/hn`

Convertible Code

Convertible Code is a one-character code used to indicate whether or not a security is eligible for conversion. N.B. this field is not yet populated.

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	-	Monthly:	-	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`convcd/0`
Array	`names[]`			Member and/or Arrays or Equivalent Usage	`names()`		*ts_print* Monthly Usage	`mconvcd/0`
Element	`convcd`			Element or Equivalent Usage	`convcd`		*stk_print* Option(s)	`/xn`

Convertible Code - Header

Convertible Code - Header is a one-character code used to indicate whether or not a security is eligible for conversion. N.B. this field is not yet populated.

Primary Concept(s)	Header Identification and Summary Data					Data Type	character
Date Range Availability and Unit of Item						Database Availability & Utility Usage	
Data Avail.	Daily: -	Monthly: -	Unit	Code		Database Format(s)	CRSPAccess
C Usage		FORTRAN-95 Usage				Product Type(s)	STK
Object	header_row	Type or Subtype	stkhdr			*ts_print* Daily Usage	n/a
Array	header	Member and/or Arrays	n/a			*ts_print* Monthly Usage	n/a
Element	hconvcd	Element	hconvcd			*stk_print* Option(s)	/hn

Count at End of Rebalancing Period

Count at End of Rebalancing Period is the count of entities belonging to a portfolio at the end of a rebalancing period. It is set to zero if unavailable.

Primary Concept(s)	Index Rebalancing History Arrays					Data Type	integer number
Date Range Availability and Unit of Item						Database Availability & Utility Usage	
Data Avail.	Daily: -	Monthly: -	Unit	Count		Database Format(s)	CRSPAccess
C Usage		FORTRAN-95 Usage				Product Type(s)	IND
Object	rebal_arr[]	Type or Subtype	rebal_arr()			*ts_print* Daily Usage	n/a
Array	rebal[][]	Member and/or Array	rebal(,)			*ts_print* Monthly Usage	n/a
Element	endcnt	Element	endcnt			*ind_print* Option(s)	/rs#

Count Available as of Rebalancing

Count Available as of Rebalancing is the total count of entities available in the universe eligible for a portfolio at the beginning of a rebalancing period. It is set to zero if unavailable.

Primary Concept(s)	Index Rebalancing History Arrays					Data Type	integer number
Date Range Availability and Unit of Item						Database Availability & Utility Usage	
Data Avail.	Daily: 1925	Monthly: 1925	Unit	Count		Database Format(s)	CRSPAccess
C Usage		FORTRAN-95 Usage				Product Type(s)	IND
Object	rebal_arr[]	Type or Subtype	rebal_arr()			*ts_print* Daily Usage	n/a
Array	rebal[][]	Member and/or Arrays	rebal(,)			*ts_print* Monthly Usage	n/a
Element	totcnt	Element	totcnt			*ind_print* Option(s)	/rs#

Count Used as of Rebalancing

Count Used as of Rebalancing is the count of entities in a portfolio as of the beginning of a rebalancing period. It is set to zero if unavailable.

Primary Concept(s)	Index Rebalancing History Arrays					Data Type	integer number
Date Range Availability and Unit of Item						Database Availability & Utility Usage	
Data Avail.	Daily: 1925	Monthly: 1925	Unit	Count		Database Format(s)	CRSPAccess
C Usage		FORTRAN-95 Usage				Product Type(s)	IND
Object	rebal_arr	Type or Subtype	rebal_arr()			*ts_print* Daily Usage	n/a
Array	rebal	Member and/or Arrays	rebal(,)			*ts_print* Monthly Usage	n/a
Element	usdcnt	Element	usdcnt			*ind_print* Option(s)	/rb#

Country Code

Country Code is a three-character code based on ISO country codes. See "Share Type" on page 125 for the list of country codes. *N.B. this field is not yet populated.*

Country Code - Header

Country Code - Header is a three-character code based on ISO country codes. N.B. this field is not yet populated.

Primary Concept(s)	Header Identification and Summary Data					Data Type	character
Date Range Availability and Unit of Item						Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Code	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`header_row`		Type or Subtype	`stkhdr`			*ts_print* Daily Usage	n/a
Array	`header`		Member and/or Arrays	n/a			*ts_print* Monthly Usage	n/a
Element	`hcntrycd`		Element	`hcntrycd`			*stk_print* Option(s)	`/hn`

CUSIP

CUSIP refers to the CUSIP identifier valid during the date range of a name structure. All non-blank *CUSIP*s are 8 characters long.

The CUSIP Agency will often change an issue's CUSIP identifier to reflect name or capital structure changes. CRSP has preserved all *CUSIP*s assigned to a given issue over time. CUSIP identifiers were first assigned in 1968. All *CUSIP*s in a name history before that date are unavailable. Dummy *CUSIP* identifiers are not included in the name history.

For more details of the CUSIP identifier; see CUSIP - Header on page 51.

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	19680102	Monthly:	19680102	Unit	Id	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`names_arr`		Type or Subtype	`names_arr`			*ts_print* Daily Usage	`ncusip/0`
Array	`names[]`		Member and/or Array	`names()`			*ts_print* Monthly Usage	`mncusip/0`
Element	`ncusip`		Element	`ncusip`			*stk_print* Option(s)	`/n`, `/xn`[1]

[1] when used with `/fs`

CUSIP - Header

CUSIP - Header is the latest 8 character CUSIP identifier for the security through the end of the file. CUSIP identifiers are supplied to CRSP by the CUSIP Service Bureau, Standard & Poor's, a division of McGraw-Hill, Inc., American Bankers Association database, Copyright 1987. See the CUSIP Copyright Information in Appendix A.6.

CUSIP identifiers were first assigned in 1968 as integers and expanded in 1984 to include alphabetic characters. The first six characters (including leading zeroes) identify the issuer, while the last two characters identify the issue. CUSIP issuer identifiers are assigned to maintain an approximately alphabetical sequence. The CUSIP identifier may change for a security if its name or capital structure changes.

No header or historical CUSIP is reused on our files. For securities no longer in existence or that were never assigned an official CUSIP identifier, CRSP has assigned a dummy CUSIP identifier for use in this field in accordance with the rules published in the CUSIP Directory. There are two potential types of dummy *CUSIP - Header*s which are assigned by CRSP. One, ***99*9*, (containing a 9 in the fourth, fifth, and seventh character positions) represents a CRSP-assigned *CUSIP* with a dummy issuer number (the first 6 character positions) and a dummy issue number (the last 2 character positions). The other, ******9*, (containing a 9 in the seventh character position) represents a CRSP-assigned *CUSIP* with a real issuer number but a dummy issue number. For example:

A *CUSIP - Header* such as 12399099 or 12345699 is assigned by CRSP, and an identifier such as 12345610 is assigned by the CUSIP Agency.

Securities actively traded on an international basis, domiciled outside the United States and Canada, will be identified by a CINS (CUSIP International Numbering System) number. CINS numbers employ the same identifier system set by the CUSIP Numbering System: Issuer (6 characters) plus Issue (2 characters) per 8 characters. It is important to note that the first portion of a CINS code is always represented by an alphabetic character, signifying the issuer's country code (domicile) or geographic region. For more information, see the current CUSIP Directory and see Share Types (Page 125).

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	`cusip/0`
Array	`header`			Member and/or Array	n/a		*ts_print* Monthly Usage	`mcusip/0`
Element	`hcusip`			Element	`hcusip`		*stk_print* Option(s)	/`hh`, /`hr`, /`hrl`, /`hn`, `hl`

Data Secondary Subtype Code

Data Secondary Subtype Code is an integer code further defining the *Data Subtype Code* of a CRSP array object. It is set to zero if unused.

Primary Concept(s)	Base CRSPAccess Data Structure(s); Event Array Objects						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_ARRAY`			Type or Subtype	`crsp_array`		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`dummy`			Element	`dummy`		*stk_print* or *ind_print* Option(s)	n/a

Data Subtype Code

Data Subtype Code is an integer code further defining categories of data in a CRSP object that otherwise have the same structure, such as the difference between a return and price data item. It is set to zero if unused. The *Data Subtype Code* for portfolio assignments and statistics time series is set to the *INDNO* of the index group with portfolio results for the market segment portfolio type.

Primary Concept(s)	Base CRSPAccess Data Structure(s); Time Series Objects, Event Array Objects, Header Objects						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_*`			Type or Subtype	`crsp_array` or `crsp_ts`		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`subtype`			Element	`subtype`		*stk_print* or *ind_print* Option(s)	n/a

Delisting Code

Delisting Code is a 3-digit integer code. It either (1) indicates that a security is still trading or (2) provides a specific reason for delisting. All coded delistings are categorized by the first digit of the delisting code.

Primary First Digit of Code	Category
1	still trading or halted but not yet delisted
2	merger
3	exchange
4	liquidation
5	delisted by NYSE, AMEX, NASDAQ or ARCA
7	delisted by the Securities and Exchange Commission
8	trading simultaneously on more than one exchange

The second and third digits of the delisting codes provide further detail of delisting events. Additional delisting codes, specific to various delisting categories, have been created to indicate when an issue is closed to further research, or if the issue is pending further research. See "6.3 Delisting Codes" on page 131 for details of delisting coding schemes.

Primary Concept(s)	Delisting History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`delist_arr`			Type or Subtype	`delist_arr`		*ts_print* Daily Usage	n/a
Array	`delist[]`			Member and/or Array	`delist()`		*ts_print* Monthly Usage	n/a
Element	`dlstcd`			Element	`dlstcd`		*stk_print* Option(s)	`/de`

Delisting Code - Header

Delisting Code - Header is the issue's delisting status at the end of the file. See *Delisting Code* for additional information.

Primary Concept(s)	Header Identification and Summary Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	n/a
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`dlstcd`			Element	`dlstcd`		*stk_print* Option(s)	`/hh`, `/hn`

Delisting Date

Delisting Date is an integer containing the date in YYYYMMDD format of a security's last price on the current exchange. If the security is still active, *Delisting Date* is set to the last date of available price data. Delisting date is never missing.

Monthly: *Delisting Date* is not necessarily a month-end trading date and might not be included in the *Calendar Trading Date* array of monthly time series.

Primary Concept(s)	Delisting History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	delist_arr			Type or Subtype	delist_arr		*ts_print* Daily Usage	n/a
Array	delist[]			Member and/or Array	delist()		*ts_print* Monthly Usage	n/a
Element	dlstdt			Element	dlstdt		*stk_print* Option(s)	/de

Delisting Date of Next Available Information

Delisting Date of Next Available Information is the integer date (in YYYYMMDD format) of a security's *Delisting Price* – the price or quote found after delisting. This date is set to zero if the security is still active. It is also set to zero if the final value of the security is determined by one or more distributions or if the value of the security is unknown after suspension of trading or after delisting.

If a liquidation or merger was announced in advance, and trading continued on the exchange, then this date is set to the date of the announcement. If the security became worthless after delisting and there is no evidence of any trading after delisting, then the *Delisting Date of Next Available Information* is set to one trading day after the *Delist Date,* and the *Delisting Price* is set to zero.

Primary Concept(s)	Delisting History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	delist_arr			Type or Subtype	delist_arr		*ts_print* Daily Usage	n/a
Array	delist[]			Member and/or Array	delist()		*ts_print* Monthly Usage	n/a
Element	nextdt			Element	nextdt		*stk_print* Option(s)	/de

Delisting Payment Date

Delisting Payment Date is the effective date (in YYYYMMDD format) of the *Amount After Delisting* value used in the *Delisting Return* calculations. If a price is used for the *Amount After Delisting,* then the *Delisting Payment Date* is set to the *Delisting Date of Next Available Information*. If distribution payments are used for the *Amount After Delisting*, then the *Delisting Payment Date* is set to the *Ex-Distribution Date* of the last known distribution payment. This date is set to zero if the security is still active or if no price or payment information is available.

Monthly: If no delisting information is found, and the security did not delist on the last trading day of the month, then the *Delisting Payment Date* is set to the *Delisting Date* and the *Amount After Delisting* is set to the last daily trading value found in the *Price or Bid/Ask Average*.

Primary Concept(s)	Delisting History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	delist_arr		Type or Subtype	delist_arr			*ts_print* Daily Usage	n/a
Array	delist[]		Member and/or Array	delist()			*ts_print* Monthly Usage	n/a
Element	dlpdt		Element	dlpdt			*stk_print* Option(s)	/de

Delisting Price

Delisting Price refers to a trade price or a price quote (given as the average of bid and ask quotes) on another exchange or over-the-counter. The date of this price or quote is specified in the *Delisting Date of Next Available Information*.

If the *Delisting Price* is positive, then it is a trade price. If the *Delisting Price* is negative, then it is the average of bid and ask quotes. A *Delisting Price* is set to zero if the security is still active, if there was no further trading for the security after the delist date, or if prices or price quotes are not available after the delist date. If delisting payments were made using distributions, the *Delisting Price* is also set to zero, and the sum of the distribution payments is specified in the *Amount After Delisting*.

Primary Concept(s)	Delisting History Array						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	USD	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	delist_arr		Type or Subtype	delist_arr			*ts_print* Daily Usage	n/a
Array	delist[]		Member and/or Array	delist()			*ts_print* Monthly Usage	n/a
Element	dlprc		Element	dlprc			*stk_print* Option(s)	/de

Delisting Return

Delisting Return is the return of a security after it has delisted from NYSE, AMEX, NASDAQ, or ARCA. The *Delisting Return* is calculated by comparing the security's *Amount After Delisting* with its price on the last day of trading. The *Amount After Delisting* can be either an off-exchange price, an off-exchange price quote, or the sum of a series of distribution payments. The effective date of the delisting return is specified in the *Delisting Payment Date*.

The return for any issue that has been closed to further research is calculated as follows:

- If a price within 10 periods of the delist date is available, then the delisting return is calculated using that price.
- If a final distribution is available, then the delisting return is calculated using all known distribution information occurring after the date of last price.
- If distributions occurring after the date of last price are available, but no final distribution has been found, then the delisting return is calculated as if a final distribution were found. (This applies only to issues closed to further research.)

- If there is evidence that no distributions will ever be paid to shareholders, then the stock is considered worthless. The delisting return is set to –1 (i.e. a 100% loss).
- If there is evidence that the stock has been declared worthless, then the delisting return is set to –1 (i.e. a 100% loss).

For any issue that is closed to further research and none of the above criteria are met, the delisting return is given a missing return code. For any issue that is pending further research, the delisting return is given a missing return code of –55.0.

Missing Delisting Return Codes

Code	Reason For Missing Return
-55.0	CRSP has no sources to establish a value after delisting or is unable to assign a value to one or more known distributions after delisting
-66.0	more than 10 trading periods between a security's last price and its first available price on a new exchange
-88.0	no data available to calculate returns
-99.0	security trades on a new exchange after delisting, but CRSP currently has no sources to gather price information

Monthly: If *Amount After Delisting* is non-zero and *Delisting Payment Date* is less than or equal to the *Delisting Date*, the *Delisting Return* represents a partial-month return, not a *Delisting Return*. The partial-month returns compare the value on the last day of trading with the value on the last month-end date and do not factor in additional after-delisting information.

Primary Concept(s)	Delisting History Array						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Ratio	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`delist_arr`			Type or Subtype	`delist_arr`		*ts_print* Daily Usage	`ret/0` when the `DLRET` option is included in the FORMAT options.
Array	`delist[]`			Member and/or Array	`delist()`		*ts_print* Monthly Usage	`mret/0` when the `DLRET` option is included in the FORMAT options.
Element	`dlret`			Element	`dlret`		*stk_print* Option(s)	`/de`

Delisting Return without Dividends

Delisting Return Without Dividends is the return of a security after it has delisted from NYSE, AMEX, NASDAQ, or ARCA. Ordinary dividends that were paid between the last trading date and the *Date of Delisting Payment* are not included in these return calculations. However, the ordinary dividends are included in the *Delisting Return* calculations. See *Delisting Return* in the Calculations Section (Page 118) for calculation and missing values.

Primary Concept(s)	Delisting History Array						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Ratio	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`delist_arr`			Type or Subtype	`delist_arr`		*ts_print* Daily Usage	`retx/0` when the `DLRET` option is included in the FORMAT options.
Array	`delist[]`			Member and/or Array	`delist()`		*ts_print* Monthly Usage	`mretx/0` when the `DLRET` option is included in the FORMAT options.
Element	`dlretx`			Element	`dlretx`		*stk_print* Option(s)	`/de`

Distribution Code

CRSP describes company distributions and corporate actions in the distribution history with a 4-digit code. The first digit describes the type of distribution. The second digit describes the payment method. The third digit augments the type denoted by the first digit. The fourth digit provides information regarding the tax status of the distribution. See "6.2 Distribution Codes" on page 126 for details.

CRSP has not verified the tax status of ordinary cash dividends since 1987. CRSP assigns the most common tax code, taxable as dividend, to ordinary dividends to these issues. CRSP does verify the tax status of stock distributions and distributions associated with rights offerings, spin-offs, liquidations, mergers, reorganizations, and exchanges.

Primary Concept(s)	Distribution Event Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`dists_arr`			Type or Subtype	`dists_arr`		*ts_print* Daily Usage	n/a
Array	`dists[]`			Member and/or Array	`dists()`		*ts_print* Monthly Usage	n/a
Element	`distcd`			Element	`distcd`		*stk_print* Option(s)	`/di`

Distribution Declaration Date

Distribution Declaration Date is the date (in `YYYYMMDD` format) on which the board of directors declared a distribution. If a declaration cannot be found, then this date is set to zero.

If the distribution is associated with a merger tender offer, then the *Distribution Declaration Date* is set to the announcement date of the tender offer. If the distribution represents merger payments or merger terms, then this date is set to the announcement date of the payments or terms.

Primary Concept(s)	Distribution Event Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`dists_arr`			Type or Subtype	`dists_arr`		*ts_print* Daily Usage	n/a
Array	`dists[]`			Member and/or Array	`dists()`		*ts_print* Monthly Usage	n/a
Element	`dclrdt`			Element	`dclrdt`		*stk_print* Option(s)	`/di`

Dividend Cash Amount

Dividend Cash Amount is the US dollar value per share of distributions resulting from cash dividends, spin-offs, mergers, exchanges, reorganizations, liquidations, and rights issues. *Dividend Cash Amount* includes the cash value of ordinary and non-ordinary (return of capital) dividends. When the distribution is paid in shares of a trading security, the *Dividend Cash Amount* is set to the exchange ratio times the price of the security at the close of the *Ex-Distribution Date*.

In a distribution where a limited percentage of shares are accepted in exchange for cash, the *Dividend Cash Amount* is set to the offer price, and the value must be adjusted using the *Factor to Adjust Price*. These are identified by a *Distribution Code* with the first digit 6 and a *Factor to Adjust Pric*e between -1 and 0. Note: regular income dividends for ADRs use the gross.

Primary Concept(s)	Distribution Event Array						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	USD	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`dists_arr`			Type or Subtype	`dists_arr`		*ts_print* Daily Usage	n/a
Array	`dists[]`			Member and/or Array	`dists()`		*ts_print* Monthly Usage	n/a
Element	`divamt`			Element	`divamt`		*stk_print* Option(s)	`/di`

Eligibility Code

Eligibility Code is a one-character code used internally by CRSP to indicate whether or not a security is included in the subscriber database. N.B. this field is not yet populated.

Primary Concept(s)	Name History Array						Data Type	Character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Code	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`names_arr`		Type or Subtype	`names_arr`			*ts_print* Daily Usage	`eligcd/0`
Array	`names[]`		Member and/or Arrays	`names()`			*ts_print* Monthly Usage	`meligcd/0`
Element	`eligcd`		Element	`eligcd`			*stk_print* Option(s)	`/xn`

Eligibility Code - Header

Eligibility Code - Header is a one-character code used internally by CRSP to indicate whether or not, and when, a security is included in the subscriber database. N.B. this field is not yet populated.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	-	Monthly:	-	Unit	Code	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`header_row`		Type or Subtype	`stkhdr`			*ts_print* Daily Usage	n/a
Array	`header`		Member and/or Arrays	n/a			*ts_print* Monthly Usage	n/a
Element	`heligcd`		Element	`heligcd`			*stk_print* Option(s)	`/hn`

End of Group Data

End of Group Data is the last date group data is valid in YYYYMMDD format.

Primary Concept(s)	Header Identification and Summary Data, Group Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`group_arr`		Type or Subtype	`group_arr()`			*ts_print* Daily Usage	n/a
Array	`group`		Member and/or Arrays	`group(,)`			*ts_print* Monthly Usage	n/a
Element	`grpenddt`		Element	`grpenddt`			*stk_print* Option(s)	`/g`

End of Stock Data

End of Stock Data is the date that data ends for the security, in `YYYYMMDD` format. It is the date of the last period in the time series arrays and is always greater than zero.

Primary Concept(s)	Header Identification and Summary Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`header_row`		Type or Subtype	`stkhdr`			*ts_print* Daily Usage	n/a
Array	`header`		Member and/or Arrays	n/a			*ts_print* Monthly Usage	n/a
Element	`enddt`		Element	`enddt`			*stk_print* or *ind_print* Option(s)	`/hh`, `/hr`, `/hrl`, `/hn`

End of Valid Data

End of Valid Data is the index of the last calendar period with valid data for a security. If no data of this type are available, it is set to zero. The *Calendar Trading Date* at this index is the date of the last calendar period with data of this type for this security.

Primary Concept(s)	Base CRSPAccess Data Structure(s); Time Series Objects						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Array index	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_TIMESERIES`			Type or Subtype	`crsp_ts`		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`end`			Element			*stk_print* or *ind_print* Option(s)	`/hr`

Exchange Code

Exchange Code is an integer code indicating the exchange on which a security is listed. *Exchange Codes* are respectively 1, 2, 3 and 4 for NYSE, AMEX, NASDAQ, and NYSE Arca. An *Exchange Code* of zero indicates that a security is either trading on an unknown exchange, or is temporarily not trading at all. Adding 30 to the normal *Exchange Codes* (31, 32, 33 and 34) identifies when-issued trading, such as during a reorganization. The following table contains a list of most of the *Exchange Codes* in the name history array. See "North American Security Exchange & Indices Codes" on page 126 for a complete list of *Exchange Codes*.

Code	Definition
-2	Halted by the NYSE, AMEX, NASDAQ or ARCA
-1	Suspended by the NYSE, AMEX, NASDAQ or ARCA
0	Not listed on exchange of current file
1	NYSE
2	AMEX
3	NASDAQ
4	NYSE Arca
31	When-issued trading on NYSE
32	When-issued trading on AMEX
33	When-issued trading on NASDAQ
34	When-issued trading on ARCA

Primary Concept(s)	Name History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`exchcd/0`
Array	`names[]`			Member and/or Array	`names()`		*ts_print* Monthly Usage	`mexchcd/0`
Element	`exchcd`			Element	`exchcd`		*stk_print* Option(s)	/n, `/xn`[1]

[1] when used with `/fs`

Exchange Code - Header

Exchange Code - Header displays the *Exchange Code* on which a security was last listed. Valid *Exchange Code - Header* values are 1, 2, 3, or 4, which correspond to the NYSE, AMEX, NASDAQ, and NYSE Arca respectively. Other *Exchange Codes* are not included in the *Exchange Code - Header* field.

Primary Concept(s)	Header Identification and Summary Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`header_row`		Type or Subtype	`stkhdr`			*ts_print* Daily Usage	
Array	`header`		Member and/or Array	n/a			*ts_print* Monthly Usage	n/a
Element	`hexcd`		Element	`hexcd`			*stk_print* Option(s)	n/a

Ex-Distribution Date

Ex-Distribution Date is the ex-dividend or ex-distribution date. It is the date on which the security is first traded without the right to receive the distribution. This date is coded as an integer in `YYYYMMDD` format and is always a daily trading date.

For distributions in a merger or exchange where the company disappeared, the *Ex-Distribution Date* is, by convention, set equal to the trading day immediately after the date of the last price.

Ex-Distribution Dates of liquidating payments after delistings are reported when available, and set to *Record Date* or *Delisting Payment Date* if unavailable.

Primary Concept(s)	Distribution Event Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`dists_arr`		Type or Subtype	`dists_arr`			*ts_print* Daily Usage	n/a
Array	`dists[]`		Member and/or Array	`dists()`			*ts_print* Monthly Usage	n/a
Element	`exdt`		Element	`exdt`			*stk_print* Option(s)	`/di`

Expiration Date

Expiration Date is the date, where applicable, that the issue expires. It is stored in YYYYMMDD format. It is set to 0 if not applicable or unknown. *N.B. this field is not yet populated.*

Expiration Date - Header

Expiration Date - Header is the date, where applicable, that the issue expires. It is stored in `YYYYMMDD` format. It is set to 0 if not applicable or unknown. N.B. this field is not yet populated.

Primary Concept(s)	Header Identification and Summary Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`header_row`		Type or Subtype	`stkhdr`			*ts_print* Daily Usage	`expdt/2`
Array	`header`		Member and/or Arrays	n/a			*ts_print* Monthly Usage	`mexpdt/2`
Element	`hexpdt`		Element	`hexpdt`			*stk_print* Option(s)	`/hn`

Factor to Adjust Price

Factor to Adjust Price is used to adjust stock prices after a distribution so that a comparison can be made on an equivalent basis between prices before and after the distribution.

Factor to Adjust Price equals *Factor to Adjust Shares Outstanding* for most distribution events. There are three types of distributions where this is the case:

1. For ordinary cash dividends or partial liquidating payments, *Factor to Adjust Price* is set to zero.
2. For cases of mergers, total liquidations, or exchanges where all shares were exchanged, a final liquidation payment was announced, or the security disappeared, *Factor to Adjust Price* is set to negative one by convention.
3. For stock dividends and splits, *Factor to Adjust Price* is the number of additional shares per old share issued:

$$facpr = \frac{s(t) - s(t')}{s(t')} = \frac{s(t)}{s(t')} - 1$$

   where *s(t)* is the number of shares outstanding, *t* is a date after or on the *Ex-Distribution Date* for the split, and *t'* is a date before the split. In a reverse split, *Factor To Adjust Price* will be between -1 and 0.

4. In other less common distribution events, spin-offs, non-total or non-final liquidating distributions, and rights, *Factor to Adjust Price* is not equal to *Factor to Adjust Shares Outstanding*. *Factor to Adjust Price* is defined as the *Dividend Cash Amoun*t divided by the stock price on the *Ex-Distribution Date*, *(P(t))*:

$$facpr = \frac{DIVAMT}{P(t)}$$

   If there is no available price on the *Ex-Distribution Date*, and there is a price within ten periods after *(P(t))*, CRSP substitutes that price for *(P(t))*.

   Note that *P(t)* is the price on the *Ex-Distribution Date*. Therefore, unless it happens to be a month-end, the price is not available in our monthly file.

5. Other cases where *Factor to Adjust Price* may not be equal to factor to adjust shares are issuances and limited tender offers. For issuances, *Factor to Adjust Price* is set to zero. For limited tender offers where a limited set percentage of shares are accepted in exchange for cash, *Factor to Adjust Price* is set to the ratio of shares accepted multiplied by -1.

Primary Concept(s)	Distribution Event Array, Calculations						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Conversion Factor	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`dists_arr`			Type or Subtype	`dists_arr`		*ts_print* Daily Usage	n/a
Array	`dists[]`			Member and/or Array	`dists()`		*ts_print* Monthly Usage	n/a
Element	`facpr`			Element	`facpr`		*stk_print* Option(s)	`/di`

Factor to Adjust Shares Outstanding

Factor to Adjust Shares Outstanding is an adjustment to *Shares Outstanding* observations due to a distribution event. It is the number of additional shares outstanding expected after the *Ex-Distribution Date* of the distribution event relative to the last known observation. *Factor to Adjust Shares Outstanding* equals *Factor to Adjust Price* for most distribution events. There are five types of distributions where this is the case. See *Factor to Adjust Price* (Page 61) for these cases and how they are handled.

For spin-offs, *Factor to Adjust Shares Outstanding* is set to zero. For rights issues, *Factor to Adjust Shares Outstanding* is calculated based on all shareholders exercising the rights on the *Ex-Distribution Date*. If it is set to 0, this distribution leaves the actual shares outstanding adjustment to this right to shares observations in the quarterly reports. For issuances and offers, if it is nonzero, then it is calculated in the same manner as for stock splits.

Primary Concept(s)	Distribution Event Array, Calculations						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Conversion Factor	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`dists_arr`			Type or Subtype	`dists_arr`		*ts_print* Daily Usage	n/a
Array	`dists[]`			Member and/or Array	`dists()`		*ts_print* Monthly Usage	n/a
Element	`facshr`			Element	`facshr`		*stk_print* Option(s)	`/di`

First Date Included in List

First Date Included in List is the date, in `YYYYMMDD` format, of the first date an issue is included in a portfolio defined as a selected list of securities.

Primary Concept(s)	Index List History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`list_arr[]`			Type or Subtype	`list_arr`		*ts_print* Daily Usage	n/a
Array	`list[][]`			Member and/or Arrays	`list(,)`		*ts_print* Monthly Usage	n/a
Element	`begdt`			Element	`begdt`		*ind_print* Option(s)	n/a

Group Flag of Associated Index

Group Flag of Associated Index identifies what group, if any, the security belongs to. There is currently only one group type, 16 - The S&P 500 Universe. If the security belongs to the selected group type, the *Group Flag of Associated Index* will contains a one (1). If it does not belong to the group, or is not valid according to the group rules, the field contains a zero (0).

Primary Concept(s)	Group Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK‡
Object	`group_arr[]`			Type or Subtype	`group_arr()`		*ts_print* Daily Usage	`grpflag1#`, `egrpflag1#`, `lgrpflag1#`
Array	`group[][]`			Member and/or Arrays	`group()`		*ts_print* Monthly Usage	`mgrpflag1#`, `megrpflag1#`, `lgrpflag1#`
Element	`grpflag`			Element	`grpflag`		*stk_print* Option(s)	`/g#`

Group Secondary Flag

Group Secondary Flag is not currently in use. It is intended to contain secondary categories needed to describe group membership. All values are set to 0.

Primary Concept(s)	Group Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	-	Monthly:	-	Unit	Code	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK‡
Object	`group_arr[]`		Type or Subtype	`group_arr()`			*ts_print* Daily Usage	n/a
Array	`group[][]`		Member and/or Arrays	`group()`			*ts_print* Monthly Usage	n/a
Element	`grpsubflag`		Element	`grpsubflag`			*stk_print* Option(s)	`/g#`

Holding Period Total Return

A return is the change in the total value of an investment in the security over some period of time per dollar of initial investment. *Holding Period Total Return* is the return for a sale on the given day. It is based on a purchase on the most recent time previous to this day when the security had a valid price. Usually, this time is the previous calendar period. See "Returns" on page 122

Daily: In daily databases, dividends are reinvested on the *Ex-Distribution Date*.

Monthly: In monthly databases, returns are holding period returns from month-end to month-end, not compounded from daily returns, and ordinary dividends are reinvested at month-end.

A series of special missing return codes specify the reason a return is missing.

Code	Reason For Missing Return
-66.0	Valid current price but no valid previous price. Either first price, unknown exchange between current and previous price, or more than 10 periods between time *t* and the time of the preceding price *t'*.
-77.0	Not trading on the current exchange at time *t*.
-88.0	No data available to calculate returns.
-99.0	Missing return due to missing price at time *t*; usually due to suspension in trading or trading on unknown exchange.

Primary Concept(s)	Price, Volume, and Return Time Series Arrays						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Ratio	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`ret_ts`		Type or Subtype	`ret_ts`			*ts_print* Daily Usage	`ret/0`
Array	`ret[]`		Member and/or Array	`ret()`			*ts_print* Monthly Usage	`mret/0`
Element			Element				*stk_print* Option(s)	`/pr, /dd, /dr, /dx`

Incorporation Code

Incorporation Code is a one-character code used to identify if the security is a domestic, foreign, or ADR security. N.B. this field is not yet populated.

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Code	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`names_arr`		Type or Subtype	`names_arr`			*ts_print* Daily Usage	`inccd/0`
Array	`names[]`		Member and/or Arrays	`names()`			*ts_print* Monthly Usage	`minccd/0`
Element	`inccd`		Element	`inccd`			*stk_print* Option(s)	`/xn`

Incorporation Code - Header

Incorporation Code - Header is a one-character code used to identify if the security is a domestic, foreign, or ADR security. N.B. this field is not yet populated.

Primary Concept(s)	Header Identification and Summary Data						Data Type	Character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	n/a
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`hinccd`			Element	`hinccd`		*stk_print* Option(s)	`/hn`

INDCO

INDCO is the permanent identifier assigned by CRSP to all groups of indices. All indices based on the same market and statistical grouping are assigned the same group number.

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	`permco/0`
Array	`indhdr`			Member and/or Array	n/a		*ts_print* Monthly Usage	`mpermco/0`
Element	`indco`			Element	`indco`		*ind_print* Option(s)	`/hh`, `/hr`

Index Basic Assignment Types Code

Index Basic Assignment Types Code is an integer code of basic assignment types for fractile or rule-based indices. The following codes are currently used.

Code	Description
0	Unknown or not applicable
1	Annual rebalancing
2	Quarterly rebalancing
3	Monthly rebalancing

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`assign`		*ts_print* Monthly Usage	n/a
Element	`assign.assigncode`			Element	`assigncode`		*ind_print* Option(s)	`/hr`

Index Basic Exception Types Code

Index Basic Exception Types Code is an integer code of the basic exception characteristics used in building an index. The following codes are currently used:

Code	Description
0	Unknown or not available
1	CRSP market index flags
2	Cap-Based index flags
3	CRSP market index trade-only prices flags

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`flags`		*ts_print* Monthly Usage	n/a
Element	`flags.flagcode`			Element	`flagcode`		*ind_print* Option(s)	`/hr`

Index Basic Rule Types Code

Index Basic Rule Types Code is an integer code of basic portfolio methodology rule types used in building indices. The following codes are currently used:

Code	Description
0	Unknown or not applicable
1	Group by previous period end issue capitalization
2	Group by previous period end company capitalization
3	Group by Scholes-Williams beta over previous year
4	Group by standard deviation over previous year

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`rules`		*ts_print* Monthly Usage	n/a
Element	`rules.rulecode`			Element	`rulecode`		*ind_print* Option(s)	`/hr`

Index Exception Handling Flags

Index Exception Handling Flags is a group of fields describing how an index supports exceptions in the data, such as new and delisted issues and missing data. The flags contain the following fields: *Index Basic Exception Type Code*, *Index New Issues Flag*, *Index Ineligible Issues Flag*, *Return of Delisted Issues Flag*, and *Index Missing Data Flag*.

Primary Concept(s)	Index Header						Data Type	structure
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Set (exception codes/flags)	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`flags`		*ts_print* Monthly Usage	n/a
Element	`flags`			Element	`n/a`		*ind_print* Option(s)	`/hr`

Index Function Code for Buy Rules

Index Function Code for Buy Rules is a code defining a function used to determine whether an issue is added to a portfolio during rebalancing. This variable is not yet available, and is always set to 0.

Primary Concept(s)	Index Header, Index Rebalancing History Arrays						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	indhdr_row			Type or Subtype	indhdr		*ts_print* Daily Usage	n/a
Array	indhdr			Member and/or Arrays	rules		*ts_print* Monthly Usage	n/a
Element	rules.buyfnct			Element	buyfnct		*ind_print* Option(s)	/hr

Index Function Code for Generating Statistics

Index Function Code for Generating Statistics is a code defining a function used to generate a statistic to be used in determining inclusion in a portfolio. The following codes are currently used.

Code	Description
0	Unknown or not applicable
1	Capitalization at end of previous period
2	Scholes-Williams beta over previous year
3	Standard deviation over previous year

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	indhdr_row			Type or Subtype	indhdr		*ts_print* Daily Usage	n/a
Array	indhdr			Member and/or Arrays	rules		*ts_print* Monthly Usage	n/a
Element	rules.statfnct			Element	statfnct		*ind_print* Option(s)	/hr

Index Function Code for Sell Rules

Index Function Code for Sell Rules is a code defining a function used to determine whether to sell current issues in a portfolio at a rebalancing period. This is not yet available, and is always set to 0.

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	indhdr_row			Type or Subtype	indhdr		*ts_print* Daily Usage	n/a
Array	indhdr			Member and/or Arrays	rules		*ts_print* Monthly Usage	n/a
Element	rules.sellfnct			Element	sellfnct		*ind_print* Option(s)	/hr

Index Group Name

Index Group Name is the name of the index group to which an index belongs. All indices with the same *Permanent Index Group Identification Number* have the same *Index Group Name*.

Primary Concept(s)	Index Header						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	IND
Object	`indhdr_row`		Type or Subtype	`indhdr`			*ts_print* Daily Usage	n/a
Array	`indhdr`		Member and/or Arrays	n/a			*ts_print* Monthly Usage	n/a
Element	`groupname`		Element	`groupname`			*ind_print* Option(s)	`/hh`, `/hr`

Index Income Return

Index Income Return is the ordinary dividend return of an index. See "Index Returns" on page 121 for details on how CRSP index returns are calculated. *Index Capital Appreciation Return* is available for CRSP-generated indices.

Primary Concept(s)	Index Time Series						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Ratio	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	IND
Object	`iret_ts[]`		Type or Subtype	`iret_ts`			*ts_print* Daily Usage	`reti`
Array	`iret[][]`		Member and/or Arrays	`iret(,)`			*ts_print* Monthly Usage	`mreti`
Element	n/a		Element	n/a			*ind_print* Option(s)	`/ir`

Index Ineligible Issues Flag

Index Ineligible Issues Flag is a code describing how issues that become ineligible for an index are handled in the index. The following codes are used:

Code	Description
0	Unknown or not available
1	Issues becoming ineligible are held until the next time period

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	IND
Object	`indhdr_row`		Type or Subtype	`indhdr`			*ts_print* Daily Usage	n/a
Array	`indhdr`		Member and/or Arrays	`flags`			*ts_print* Monthly Usage	n/a
Element	`flags.delflag`		Element	`delflag`			*ind_print* Option(s)	`/hr`

Index Level of Capital Appreciation

Index Capital Appreciation Return Index Level is the value of an index, excluding ordinary dividends, relative to its value at one fixed point in time. See "Index Levels" on page 121 for details on how CRSP index levels are calculated. If CRSP includes a public index such as the S&P 500 Composite or the NASDAQ Composite, *Index Capital Appreciation Return Index Level* is maintained and provided directly by the creator of the index.

Primary Concept(s)	Index Time Series						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	USD	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`aind_ts[]`			Type or Subtype	`aind_ts`		*ts_print* Daily Usage	`aind/0`
Array	`aind[][]`			Member and/or Arrays	`aind(,)`		*ts_print* Monthly Usage	`maind/0`
Element	n/a			Element	n/a		*ind_print* Option(s)	`/ai`

Index Level of Returns on Income

Index Income Return Index Level is the ordinary dividend value of an index, relative to its value at one fixed point in time. See "Index Levels" on page 121 for details on how CRSP index levels are calculated. *Index Income Return Index Level* is available for CRSP-generated indices.

Primary Concept(s)	Index Time Series						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	USD	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`iind_ts[]`			Type or Subtype	`iind_ts`		*ts_print* Daily Usage	`iind`
Array	`iind[][]`			Member and/or Arrays	`iind(,)`		*ts_print* Monthly Usage	`miind`
Element	n/a			Element	n/a		*ind_print* Option(s)	`/ii`

Index Method Type Code

Index Method Type Code is an integer code indicating the basic methodology for an index. It represents the combination of *Index Primary Methodology Type, Index Secondary Methodology Group*, *Index Reweighting Type Flag*, and *Index Reweighting Timing Flag* characteristics. See "Chapter 3: CRSP Index Methodologies" on page 27 Current codes are:

Code	Description
1	CRSP Cap-Based Portfolios
3	CRSP Risk-Based Decile Indices
4	CRSP Value-Weighted Market Indices
5	CRSP Equal-Weighted Market Indices
6	CRSP Capitalization Decile Market Indices
7	S&P 500® Composite
8	CRSP Value-Weighted Index on the S&P 500® Universe
9	CRSP Equal-Weighted Index on the S&P 500® Universe
10	NASDAQ Composite
12	CRSP Fixed Term Bond Returns
13	CRSP Fixed Term Bill Returns
14	Provided by External Source

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`method`		*ts_print* Monthly Usage	n/a
Element	`method.methcode`			Element	`methcode`		*ind_print* Option(s)	`/hr`

Index Methodology Description Structure

Index Methodology Description Structure contains fields describing the rules used to build the index. These fields contain information on primary and secondary methodologies and rules for weighting securities within the index. The fields are *Index Method Type Code, Index Primary Methodology Type, Index Secondary Methodology Group*, *Index Reweighting Type Flag*, and *Index Reweighting Timing Flag.*

Primary Concept(s)	Index Header						Data Type	structure
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Set (methodology)	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`method`		*ts_print* Monthly Usage	n/a
Element	`method`			Element	n/a		*ind_print* Option(s)	`/hr`

Index Missing Data Flag

Index Missing Data Flag describes the possible actions taken for securities with missing data during the range in an index portfolio. The following codes are currently used:

Code	Description
0	Unknown or not applicable
3	Issues without single period returns are excluded
5	Alternate prices are used if possible to generate single period returns
13	Quotes without trades are treated as missing prices

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`flags`		*ts_print* Monthly Usage	n/a
Element	`flags.missflag`			Element	`missflag`		*stk_print* or *ind_print* Option(s)	`/hr`

Index Name

Index Name is the name of the index or portfolio. The *Index Names* are listed in the *INDNO* table. See "3.6 CRSP Index Series and Groups" on page 33 for a complete list of Index Names.

Primary Concept(s)	Index Header						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`indname`			Element	`indname`		*ind_print* Option(s)	`/hh`, `/hr`

Index New Issues Flag

Index New Issues Flag is an integer code describing how new issues are used in an index. The following codes are used:

Code	Description
0	Unknown or not available
1	New securities are included the first period they meet existing portfolio restrictions
2	Securities are never added until next rebalancing period

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`flags`		*ts_print* Monthly Usage	n/a
Element	`flags.addflag`			Element	`addflag`		*ind_print* Option(s)	`/hr`

Index Primary Link

Index Primary Link is the *INDNO* of an index group containing this index portfolio series. It is set to zero if this index is a group or if there is no primary group index associated with this index series. A series index representing one portfolio of a group can use *Index Primary Link* to refer back to the primary index. The primary index contains rebalancing information and data for all portfolios in that group.

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`primflag`			Element	`primflag`		*ind_print* Option(s)	`/hh`, `/hr`

Index Primary Methodology Type

Index Primary Methodology Type is an integer code describing the index type. The following types are currently used:

Code	Name	Description
0	Fractile Index	A market segment index where breakpoints based on some rule and/or statistic are used to divide eligible issues into portfolios at different intervals. The breakpoint function is continuous so that all eligible issues are in exactly one portfolio during each period.
1	Selected Index	Universe is supplied from an outside source, with given issues or companies and the data ranges for each.
3	Market Index	Portfolio of all eligible issues is reevaluated each period based on constant universe restrictions.
4	Other	Not Applicable.

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`method`		*ts_print* Monthly Usage	n/a
Element	`method.primtype`			Element	`primtype`		*ind_print* Option(s)	`/hr`

Index Rebalancing Begin Date

Index Rebalancing Begin Date is the integer date, in `YYYYMMDD` format, of the first date in the rebalancing period of an index.

Primary Concept(s)	Index Rebalancing History Arrays						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`rebal_arr[]`			Type or Subtype	`rebal_arr()`		*ts_print* Daily Usage	n/a
Array	`rebal[][]`			Member and/or Arrays	`rebal(,)`		*ts_print* Monthly Usage	n/a
Element	`rbbegdt`			Element	`rbbegdt`		*ind_print* Option(s)	`/rb#`

Index Rebalancing End Date

Index Rebalancing End Date is the integer date, in YYYYMMDD format, of the last date in the rebalancing period of an index.

Primary Concept(s)	Index Rebalancing History Arrays						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage\			FORTRAN-95 Usage				Product Type(s)	IND
Object	rebal_arr[]		Type or Subtype	rebal_arr()			*ts_print* Daily Usage	n/a
Array	rebal[][]		Member and/or Arrays	rebal(,)			*ts_print* Monthly Usage	n/a
Element	rbenddt		Element	rbenddt			*ind_print* Option(s)	/rb#

Index Returns without Dividends

Index Capital Appreciation Return is the return, excluding ordinary dividends, of an index. See "Index Returns" on page 121 for details on how CRSP index returns are calculated. If CRSP includes a public index such as the S&P 500 Composite or the NASDAQ Composite, *Index Capital Appreciation Return* is derived from data provided by the creator of the index.

Primary Concept(s)	Index Time Series						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Ratio	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK*, IND
Object	aret_ts[]		Type or Subtype	aret_ts			*ts_print* Daily Usage	retx/0
Array	aret[][]		Member and/or Arrays	aret(,)			*ts_print* Monthly Usage	mretx/0
Element	n/a		Element	n/a			*ind_print* Option(s)	/ar

Index Reweighting Timing Flag

Index Reweighting Timing Flag is an integer code indicating how frequently weights are recalculated in the existing portfolio. The following codes are currently used:

Code	Description
0	Not available
11	Weights are applied each time period

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	IND
Object	indhdr_row		Type or Subtype	indhdr			*ts_print* Daily Usage	n/a
Array	indhdr		Member and/or Arrays	method			*ts_print* Monthly Usage	n/a
Element	method.wgtflag		Element	wgtflag			*ind_print* Option(s)	/hr

Index Reweighting Type Flag

Index Reweighting Type Flag is an integer code indicating the method of weighting the issues in the portfolio index. The following codes are currently used:

Code	Description
0	Not available
1	Value-weighted, weights not supplied by CRSP
2	Value-weighted
3	Equal-weighted

Primary Concept(s)	Index Header, Portfolio Statistics & Assignments Arrays						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`method`		*ts_print* Monthly Usage	n/a
Element	`method.wgttype`			Element	`wgttype`		*ind_print* Option(s)	`/hr`

Index Secondary Methodology Group

Index Secondary Methodology Group is an integer code with further detail for the *Index Primary Methodology Type*. The following codes are currently used:

Code	Description
0	No further description
10	Portfolios based on market capitalization
12	Portfolios based on result statistic: beta or standard deviation
13	Issues in S&P 500® Index
14	Issues in the NASDAQ Composite Index
15	Treasury Issues of Selected Maturity Ranges

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`method`		*ts_print* Monthly Usage	n/a
Element	`method.subtype`			Element	`subtype`		*ind_print* Option(s)	`/hr`

Index Statistic Grouping Code

Index Statistic Grouping Code is an integer code describing the type of grouping done on issues before any statistics are applied. The following codes are currently used:

Code	Description
0	Unknown or not applicable
1	Each issue is grouped independently
2	Multiple issues of a company are combined

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`rules`		*ts_print* Monthly Usage	n/a
Element	`rules.groupflag`			Element	`groupflag`		*stk_print* or *ind_print* Option(s)	`/hr`

Index Subcategory Code

Index Subcategory Code is an integer flag indicating a subcategory of the primary index in an index list history to which the security belongs. It is set to zero if no subcategory is applicable.

Primary Concept(s)	Index List History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`list_arr[]`			Type or Subtype	`list_arr()`		*ts_print* Daily Usage	n/a
Array	`list[][]`			Member and/or Arrays	`list(,)`		*ts_print* Monthly Usage	n/a
Element	`subind`			Element	`subind`		*ind_print* Option(s)	n/a

Index Subset Screening Structure

Index Subset Screening Structure, like the *Partition Subset Screening Structure*, is a structure of fields used to restrict a database using various screening variables. The screen fields are: *Universal Subset Type Code*, *First Trading Date Allowed in Restriction*, *Index Restriction End Date*, *Valid Exchange Codes in Universe*, *Valid NASDAQ Market Groups in Universe*, *Valid When-Issued Securities in Universe*, *Valid Incorporation of Securities in Universe*, and *Share Code Screen Structure*. *Index Subset Screening Structure* screens are used to restrict the securities used in the actual index.

Primary Concept(s)	Index Header						Data Type	structure
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Set (screen markets)	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Array	`induniv`		*ts_print* Monthly Usage	n/a
Element	`induniv`			Element	n/a		*ind_print* Option(s)	`/hr`

Index Total Count

Index Total Count is the total number of securities in the index universe with a valid price on the selected *Calendar Trading Date*. See "Chapter 3: CRSP Index Methodologies" on page 27 for information including rebalancing frequency and universe inclusion for specific indexes.

Primary Concept(s)	Index Time Series						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Count	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK*, IND
Object	totcnt_ts[]			Type or Subtype	totcnt_ts		*ts_print* Daily Usage	cnt
Array	totcnt[][]			Member and/or Arrays	totcnt(,)		*ts_print* Monthly Usage	mcnt
Element	n/a			Element	n/a		*ind_print* Option(s)	/tc

Index Total Return

Index Total Return is the return, including all distributions, of an index. See "Index Returns" on page 121 for details on how CRSP index returns are calculated. *Index Total Return* is only available for CRSP-generated indexes.

Primary Concept(s)	Index Time Series						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Ratio	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK*, IND
Object	tret_ts[]			Type or Subtype	tret_ts		*ts_print* Daily Usage	n/a
Array	tret[][]			Member and/or Arrays	tret(,)		*ts_print* Monthly Usage	n/a
Element	n/a			Element	n/a		*ind_print* Option(s)	/tr

Index Level of Total Returns

Index Total Return Index Level is the value of an index, including all distributions, relative to its value at one fixed point in time. See "Index Levels" on page 121 for details on how CRSP index levels are calculated. *Index Total Return Index Level* is only available for CRSP-generated indexes. Index levels for the CRSP Stock File Indices and the CRSP CTI Indices are set to an initial value of 100.00 on 19721229. Index levels for the Cap-Based Portfolios are set to 1.00 on 19251231.

Primary Concept(s)	Index Time Series						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	USD	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	tind_ts[]			Type or Subtype	tind_ts		*ts_print* Daily Usage	tind
Array	tind[][]			Member and/or Arrays	tind(,)		*ts_print* Monthly Usage	mtind
Element	n/a			Element	n/a		*ind_print* Option(s)	/ti

Index Total Value

Index Total Value is the total market value of the non-ADR securities in the index universe, in $1000s, with valid prices and shares outstanding amounts on the selected *Calendar Trading Date*. See "Chapter 3: CRSP Index Methodologies" on page 27 for information including rebalancing frequency and universe inclusion for specific indexes.

Primary Concept(s)	Index Time Series						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	USD	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK*, IND
Object	`totval_ts[]`			Type or Subtype	`totval_ts`		*ts_print* Daily Usage	`cap/0`
Array	`totval[][]`			Member and/or Arrays	`totval(,)`		*ts_print* Monthly Usage	`mcap/0`
Element	n/a			Element	n/a		*ind_print* Option(s)	`/tv`

Index Used Count

Index Used Count contains the count of issues used to create index results for a specific index or portfolio during one calendar period. A security must be a member of the index or portfolio with valid prices for both the current and the previous trading periods to be included in the count. See "Chapter 3: CRSP Index Methodologies" on page 27 for information including rebalancing frequency and universe inclusion for specific indexes.

Primary Concept(s)	Index Time Series						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Count	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK*, IND
Object	`usdcnt_ts[]`			Type or Subtype	`usdcnt_ts`		*ts_print* Daily Usage	`cnt/1`
Array	`usdcnt[][]`			Member and/or Arrays	`usdcnt(,)`		*ts_print* Monthly Usage	`mcnt/1`
Element	n/a			Element	n/a		*ind_print* Option(s)	`/uc`

Index Used Value

Index Used Value is the beginning total market value, in $1000s, of all securities that are used in an index on the selected *Calendar Trading Date*. In a CRSP value-weighted index the *Index Used Value* is the weight of the index.

For standard CRSP market indices the beginning total market value is calculated using prices and shares from the previous trading day. In these indices a security cannot be an ADR and must have prices and shares on the current and previous trading dates. See "Index Returns" on page 121, and see Chapter 3: CRSP Index Methodologies on page 27.

Primary Concept(s)	Index Time Series						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	USD	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK*, IND
Object	`usdval_ts[]`			Type or Subtype	`usdval_ts`		*ts_print* Daily Usage	`cap/1`
Array	`usdval[][]`			Member and/or Arrays	`usdval(,)`		*ts_print* Monthly Usage	`mcap/1`
Element	n/a			Element	n/a		*ind_print* Option(s)	`/uv`

INDNO

INDNO indicates the unique permanent identifier assigned by CRSP to every supported index. All *INDNO* identifiers are 7-digit integers. There is no inherent meaning in the numbers. The indices sets in a CRSPAccess database are sorted by this field. See "3.6 CRSP Index Series and Groups" on page 33 for a full list of CRSP Indices.

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	`permno`
Array	`indhdr`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	`mpermno`
Element	`indno`			Element	`indno`		*ind_print* Option(s)	`/hh, /hr`

INDNO of Associated Index

INDNO of Associated Index is the identifier of an associated index used to supply rebalancing breakpoint information used for assignments or buy/sell rules to this index. It is set to zero if external portfolio data are used.

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`assign`		*ts_print* Monthly Usage	n/a
Element	`assign.asperm`			Element	`asperm`		*ind_print* Option(s)	`/hr`

Interest Rate or Strike Price - Header

Interest Rate of Strike Price - Header is the guaranteed annualized interest rate, in percentages, for preferred stock or effective rate of a right, warrant or debt hybrid. It is set to 0 if not applicable or not known.

Primary Concept(s)	Header Identification and Summary Data						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	% rate	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	n/a
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`hrating`			Element	`hrating`		*stk_print* Option(s)	`/hn`

Intermarket Trading System Indicator

Intermarket Trading System Indicator is a one-character flag that indicates whether the security is eligible to trade on the intermarket system. N.B. this field is not yet populated.

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`its/0`
Array	`names[]`			Member and/or Arrays	`names()`		*ts_print* Monthly Usage	`mits/0`
Element	`its`			Element	`its`		*stk_print* Option(s)	`/xn`

Intermarket Trading System Indicator - Header

Intermarket Trading System Indicator - Header is a one-character flag that indicates whether the security is eligible to trade on the intermarket system. N.B. this field is not yet populated.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	n/a
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`hits`			Element	`hits`		*stk_print* Option(s)	`/hn`

Issuer Code

Issuer Code is a one character code that identifies the organizational type of a security. N.B. this field is not yet populated.

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`issuercd/0`
Array	`names[]`			Member and/or Arrays	`names()`		*ts_print* Monthly Usage	`missuercd/0`
Element	`issuercd`			Element	`issuercd`		*stk_print* Option(s)	`/xn`

Issuer Code - Header

Issuer Code - Header is a one character code that identifies the organizational type of a security. N.B. this field is not yet populated.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	n/a
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`hissuercd`			Element	`hissuercd`		*stk_print* Option(s)	`/hn`

Last Date Included in List

Last Date Included in a List is the integer date, in `YYYYMMDD` format, of the last date an issue is included in a portfolio, defined as a time-dependent list of members.

Primary Concept(s)	Index List History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`list_arr[]`			Type or Subtype	`list_arr()`		*ts_print* Daily Usage	n/a
Array	`list[][]`			Member and/or Arrays	`list_arr(,)`		*ts_print* Monthly Usage	n/a
Element	`enddt`			Element	`enddt`		*ind_print* Option(s)	`/li`

Last Date of Name

Last Date of Name is the last effective date of a security's name history structure. It is set to the date preceding the *Name Effective Date* of the next name structure, the maximum of *End of Stock Data*, or the *Delisting Date* of the last name structure.

The name information on any given date can be found by finding the name structure where the target date is between *Name Effective Date* and *Last Date of Name*. There is always only one name structure for any select date between the first *Name Effective Date* and the *Delisting Date*.

Primary Concept(s)		Name History Array					Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	n/a
Array	`names[]`			Member and/or Arrays	`names()`		*ts_print* Monthly Usage	n/a
Element	`nameenddt`			Element	`nameenddt`		*stk_print* Option(s)	/n, `/xn`[1], `/an`[1]

[1] When used with `/fs`.

Maximum Count During Period

Maximum Count During Period is the largest count of issues in a portfolio at any point within an index rebalancing period.

Primary Concept(s)		Index Rebalancing History Arrays					Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Count	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`rebal_arr[]`			Type or Subtype	`rebal_arr()`		*ts_print* Daily Usage	n/a
Array	`rebal[][]`			Member and/or Arrays	`rebal(,)`		*ts_print* Monthly Usage	n/a
Element	`maxcnt`			Element	`maxcnt`		*ind_print* Option(s)	`/rs#`

Maximum Number of Array Elements

Maximum Number of Array Elements is the maximum number of time periods available in a time series or calendar, or the maximum number of observations in an event array.

Primary Concept(s)		Base CRSPAccess Data Structure(s);Time Series Objects, Event Array Objects, Calendar Objects					Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Metadata	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_TIMESERIES`, `CRSP_CAL`, `CRSP_ARRAY`			Type or Subtype			*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays			*ts_print* Monthly Usage	n/a
Element	`maxarr`			Element			*stk_print* or *ind_print* Option(s)	n/a

Name Code

N.B. this field is not yet populated. *Name Code* is reserved.

Name Code - Header

Name Code - Header is reserved. N.B. this field is not yet populated.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Code	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`header_row`		Type or Subtype	`stkhdr`			*ts_print* Daily Usage	n/a
Array	`header`		Member and/or Arrays	n/a			*ts_print* Monthly Usage	n/a
Element	`hnamecd`		Element	`hnamecd`			*stk_print* Option(s)	`/hn`

Name Description

N.B. this field is not yet populated. *Name Description* is a a 24-character reserved field.

Name Description - Header

Name Description - Header is a 24-character reserved field. N.B. this field is not yet populated.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Description	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`header_row`		Type or Subtype	`stkhdr`			*ts_print* Daily Usage	n/a
Array	`header`		Member and/or Arrays	n/a			*ts_print* Monthly Usage	n/a
Element	`hnamedesc`		Element	`hnamedesc`			*stk_print* Option(s)	`/hn`

Name Effective Date

Name Effective Date is the starting date of a set of security name information, stored in `YYYYMMDD` format. If the *CUSIP*, *Company Name*, *Ticker Symbol*, *Exchange Code*, or *s* changes, CRSP adds a new name structure that records the change and the date the change became effective. *Name Effective Date* is the date associated with a specific name structure.

Primary Concept(s)	Name History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`names_arr`		Type or Subtype	`names_arr`			*ts_print* Daily Usage	n/a
Array	`names[]`		Member and/or Array	`names()`			*ts_print* Monthly Usage	n/a
Element	`namedt`		Element	`namedt`			*stk_print* Option(s)	`/n`, `/xn`

Name Flag

Name Flag is a reserved one-character code. N.B. this field is not yet populated.

Primary Concept(s)	Name History Array	Data Type	character

Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Code	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	names_arr		Type or Subtype	names_arr			*ts_print* Daily Usage	nameflag/0
Array or Equivalent Usage	names[]		Member and/or Arrays	names()			*ts_print* Monthly Usage	mnameflag/0
Element or Equivalent Usage	nameflag		Element	nameflag			*stk_print* Option(s)	/xn[1]

[1] When used with /fs.

Name Flag - Header

Name Flag - Header is a reserved one-character code. N.B. this field is not yet populated.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Code	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	header_row		Type or Subtype	stkhdr			*ts_print* Daily Usage	n/a
Array	header		Member and/or Arrays	n/a			*ts_print* Monthly Usage	n/a
Element	hnameflag		Element	hnameflag			*stk_print* Option(s)	/hn

NASD Index Code

NASD Index Code is an integer code indicating the issue's classification within NASD's internal business description categories. This field is not available between April, 1998 and February, 2000. The following codes are used in the file:

Code	Description	Code	Description
0	Unknown or unavailable	30	NASDAQ NNM Biotechnology
1	No index	31	NASDAQ NNM Composite
2	NASDAQ Industrial	42	NASDAQ SCM Industrial
3	NASDAQ Bank	43	NASDAQ SCM Bank
4	NASDAQ Other Financial Institution	44	NASDAQ SCM Other Financial Institution
5	NASDAQ Insurance Company	45	NASDAQ SCM Insurance
6	NASDAQ Transportation company	46	NASDAQ SCM Transportation
7	NASDAQ Utility Company	47	NASDAQ SCM Utility
8	NASDAQ Telecommunication	48	NASDAQ SCM Telecommunication
9	NASDAQ Computer	49	NASDAQ SCM Computer
10	NASDAQ Biotechnology	50	NASDAQ SCM Biotechnology
11	NASDAQ Composite	51	NASDAQ SCM Composite
12	NASDAQ 100	62	NASDAQ Financial 100
13	NASDAQ 100 (Unadjusted)	69	NASDAQ Computer 100
14	Generic X	81	Wilder NASDAQ OMX Global Energy Efficient Transport Index (symbol: HAUL)
15	Generic Z	82	Wilder NASDAQ OMX Global Energy Efficient Transport Total Return Index (symbol: HAUX)
16	Generic Y	83	NASDAQ OMX Global Biotechnology Index (symbol: QGBI)
22	NASDAQ NNM Industrial	84	NASDAQ OMX Global Biotechnology Total Return Index (symbol: QGBX)
23	NASDAQ NNM Bank	85	NASDAQ OMX Global Coal Index (symbol: QCOL)
24	NASDAQ NNM Other Financial Institution	86	NASDAQ OMX Global Coal Total Return Index (symbol: QCLX)
25	NASDAQ NNM Insurance	87	NASDAQ OMX Global Steel Index (symbol: QSTL)
26	NASDAQ NNM Transportation	88	NASDAQ OMX Global Steel Total Return Index (symbol: QSTS)
27	NASDAQ NNM Utility	89	NASDAQ OMX Global Gold & Precious Metals Index (symbol: QGLD)
28	NASDAQ NNM Telecommunication	90	NASDAQ OMX Global Gold & Precious Metals Total Return Index (symbol: QGLX)
29	NASDAQ NNM Computer		

Primary Concept(s)	NASDAQ Information Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	198211	Monthly:	198211	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`nasdin_arr`			Type or Subtype	`nasdin_arr`		*ts_print* Daily Usage	`nsdinx/0`
Array	`nasdin[]`			Member and/or Array	`nasdin()`		*ts_print* Monthly Usage	`mnsdinx/0`
Element	`nsdinx`			Element	`nsdinx`		*stk_print* Option(s)	`/q`

NASDAQ Company Number

NASDAQ Company Number is a unique integer assigned by the National Association of Securities Dealers (NASD) to each company with a listed security on The NASDAQ Stock Market[SM]. If the company never traded an issue on The NASDAQ Stock Market[SM], or is unknown, *NASDAQ Company Number* is set to 0. The *NASDAQ Company Number* may change if NASDAQ assigns a new number to an issue that CRSP considers to be the same company as a company that previously traded on the NASDAQ Stock Market[SM].

Primary Concept(s)	Header Identification and Summary Data,						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	198211	Monthly:	198211	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	`compno/0`
Array	`header`			Member and/or Array	n/a		*ts_print* Monthly Usage	`mcompno/0`
Element	`compno`			Element	`compno`		*stk_print* Option(s)	`/hh`, `/hr`, `/hrl`, `/hn`

NASDAQ Issue Number

NASDAQ Issue Number is a unique integer assigned by the National Association of Securities Dealers (NASD) to each listed security on The NASDAQ Stock Market[SM]. It is this issue-specific identifier which differentiates securities issued by the same company. If the issue number is unknown, the *NASDAQ Issue Number* is set to zero. If an NYSE/AMEX security was ever traded on NASDAQ, this number is set to the last issue number assigned when it was trading on NASDAQ. The *NASDAQ Issue Number* in the CRSP Data File may change if NASDAQ assigns a new number to an issue CRSP considers to be a continuation of an existing issue.

Primary Concept(s)	Header Identification and Summary Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	198211	Monthly:	198211	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	n/a
Array	`header`			Member and/or Array	n/a		*ts_print* Monthly Usage	n/a
Element	`issuno`			Element	`issuno`		*stk_print* Option(s)	`/hh, /hr, /hrl, /hn`

NASDAQ Market Makers

Market Maker Count is the number of active market makers for an issue trading on the NASDAQ Stock Market. This contains a 0 if there are no active market makers at that time, or if the date falls in December of 1982 for a NASD Company Number less than 1025, or in February of 1986.

Primary Concept(s)	NASDAQ Information Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	19821101	Monthly:	198211	Unit	Count	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`nasdin_arr`			Type or Subtype	`nasdin_arr`		*ts_print* Daily Usage	`mmcnt/0`
Array	`nasdin[]`			Member and/or Array	`nasdin()`		*ts_print* Monthly Usage	`mmmcnt/0`
Element	`mmcnt`			Element	`mmcnt`		*stk_print* Option(s)	`/q`

NASDAQ National Market Indicator

NASDAQ National Market Indicator is a 1-digit integer code indicating whether or not an issue is a member of The NASDAQ National Market (formerly NMS). Prior to June 15, 1992, transaction data was not available for NASDAQ SmallCap Securities. As of June 15, 1992 transaction data became available.

NASDAQ introduced a 3-tier market initiative in July 2006. As a result, the CRSP NASDAQ National Market Indicator (NMSIND) coding scheme was changed. Specifically:

- Small-Cap - renamed Capital Market, CRSP code 4
- National Market - split into two:
  - Global Market, CRSP code 4
  - Global Select Market, CRSP code 6.

To achieve expected results, the subset functionality in the ts_print interface is labeled to reflect these changes. After July 1, 2006, SmallCap is renamed to Capital Market. National Market is renamed to Global Market. In addition, a subset of Global Market is identified as the Global Select Market.

Code	Description
0	Unknown or unavailable
1	The NASDAQ SmallCap Market before June 15, 1992
2	The NASDAQ National Market
3	The NASDAQ SmallCap Market after June 15, 1992
4	Capital Market (formerly SmallCap) after July 1, 2006
5	Global Market (formerly National Market) after July 1, 2006
6	Global Select Market - new subset of Global Market after July 1, 2006

Primary Concept(s)	NASDAQ Information Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	19920615	Monthly:	199206	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	nasdin_arr			Type or Subtype	nasdin_arr		*ts_print* Daily Usage	nmsind/0
Array	nasdin[]			Member and/or Array	nasdin()		*ts_print* Monthly Usage	mnmsind/0
Element	nmsind			Element	nmsind		*stk_print* Option(s)	/q

NASDAQ Number of Trades

Daily: *NASDAQ Number of Trades* contains the number of trades made on the NASDAQ Stock Market each date for a security. Trades on all exchanges are connected to NASDAQ's composite pricing network and all late trades are included in the count. If the number of trades is unavailable, the field is set to –99.

Number of trades is available only for issues trading on The NASDAQ Stock Market[SM]. It is reported for all securities listed on The NASDAQ National Market since November 1, 1982, and all NASDAQ securities since June 15, 1992. Due to lack of sources, *NASDAQ Number of Trades* data are missing for 15 NASDAQ National Market securities in December, 1982, and all The NASDAQ National Market securities in February, 1986.

Monthly: Not available.

Primary Concept(s)	Supplemental NASDAQ Time Series						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	19821101	Monthly:	n/a	Unit	Count	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`numtrd_ts`			Type or Subtype	`numtrd_ts`		*ts_print* Daily Usage	`numtrd/0`
Array	`numtrd[]`			Member and/or Array	`numtrd()`		*ts_print* Monthly Usage	n/a
Element	`numtrd`			Element	`numtrd`		*stk_print* Option(s)	`/pn`

NASDAQ Status Code

NASDAQ Status Code is a 1-digit integer code describing the trading status of an issue listed on the NASDAQ Stock Market. This code was in limited use between April, 1998 and May 2000. During that time only codes 1 (active) and 5 (delisted) were assigned by CRSP. Codes used in the files include:

Code	Description
0	Unknown or not applicable
1	Active
2	Trading with only one market maker
3	Suspended
4	Inactive
5	Delisted

Primary Concept(s)	NASDAQ Information Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	19821101	Monthly:	198211	Unit	Code	Database Format(s)	`CRSPAccess`
C Usage				FORTRAN-95 Usage			Product Type(s)	`STK`
Object	`nasdin_arr`			Type or Subtype	`nasdin_arr`		*ts_print* Daily Usage	`trtscd/0`
Array	`nasdin[]`			Member and/or Array	`nasdin()`		*ts_print* Monthly Usage	`mtrtscd/0`
Element	`trtscd`			Element	`trtscd`		*stk_print* Option(s)	`/q`

NASDAQ Status Date

NASDAQ Status Date is the effective integer begin date, in `YYYYMMDD` format, for a NASDAQ information structure for a security listed on the NASDAQ Stock Market.

Primary Concept(s)	NASDAQ Information Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	19821101	Monthly:	198211	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`nasdin_arr`			Type or Subtype	`nasdin_arr`		*ts_print* Daily Usage	n/a
Array	`nasdin[]`			Member and/or Array	`nasdin()`		*ts_print* Monthly Usage	n/a
Element	`trtsdt`			Element	`trtsdt`		*stk_print* Option(s)	`/q`

NASDAQ Traits End Date

NASDAQ Traits End Date is the last date, in `YYYYMMDD` format, for which information in a NASDAQ information array structure is valid for a security. It is set to the last trading date before the next NASDAQ information event, or to 99999999 in the last structure.

Primary Concept(s)	NASDAQ Information Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	19821101	Monthly:	198211	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`nasdin_arr`			Type or Subtype	`nasdin_arr`		*ts_print* Daily Usage	n/a
Array	`nasdin[]`			Member and/or Arrays	`nasdin()`		*ts_print* Monthly Usage	n/a
Element	`trtsenddt`			Element	`trtsenddt`		*stk_print* Option(s)	`/q`

New PERMCO

New PERMCO is an integer link to a company assigned when an issue ceases trading as a result of a merger or exchange when shareholders receive some payment from the acquiring company. If *New* PERMNO is nonzero, *New* PERMCO is the PERMCO of that security. If *New* PERMNO is zero, *New* PERMCO can still be nonzero if the shareholders receive a payment from an acquiring company known to CRSP, but the payments are not primarily in the stock of the company. *New PERMCO* is zero if the company is unknown to CRSP or if the delisting does not represent a merger or exchange. See *Acquiring PERMCO* for companies associated with individual payments.

Primary Concept(s)	Delisting History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`delist_arr`			Type or Subtype	`delist_arr`		*ts_print* Daily Usage	n/a
Array	`delist[]`			Member and/or Array	`delist()`		*ts_print* Monthly Usage	n/a
Element	`nwcomp`			Element	`nwcomp`		*stk_print* Option(s)	`/de`

New PERMNO

New PERMNO is an integer pointer to a new security assigned when an issue ceases trading as a result of a merger or exchange where shareholders receive stock in the acquiring company. The *New* PERMNO is the PERMNO of the primary security received from the acquiring company. It acts as a forward pointer, allowing the user to trace the ongoing history of surviving companies. *New* PERMNO may identify an issue that exists on a different CRSP Stock File. It is set to zero if there is no new primary security applicable, the issue is unknown, or the delisting does not represent a merger or exchange. The distribution history arrays contain an itemized record of all types of payments to shareholders in an exchange or merger. See *Acquiring PERMNO* for companies associated with individual payments.

Primary Concept(s)	Delisting History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`delist_arr`			Type or Subtype	`delist_arr`		*ts_print* Daily Usage	n/a
Array	`delist[]`			Member and/or Array	`delist()`		*ts_print* Monthly Usage	n/a
Element	`nwperm`			Element	`nwperm`		*stk_print* Option(s)	`/de`

North American Industry Classification System Code

North American Industry Classification System Code (NAICS) is an 6-character code used to group companies with similar products or services. It was adopted in 1997 and implemented in 1999, by the Office of Management and Budget (OMB), to replace the U.S. Standard Industrial Classification (SIC) system. They are assigned to companies' establishments with similar products or services. While there are exceptions to the definition, an establishment "is generally a single, physical location at which economic activity occurs (e.g., store, factory, farm, etc.)[1]". NAICS was designed to encompass all fields of economic activities, producing and non-producing. Each establishment is assigned to one industry that matches its primary activity. The codes were developed by the US, Canada, and Mexico to provide a business activity standard throughout North America, to facilitate economic analyses of North America's economies.

NAICS is a hierarchical code, containing up to six digits: The first two fields, NAICS sectors, designate general categories of economic activity, the third field, sub-sector, further defines the sector, the fourth field is the industry group, the fifth field is the NAICS industry, and the sixth field represents the national industry (a zero in the 6th digit generally indicates that the NAICS industry and the country industry are the same). For example, 1123 represents Poultry and Egg Production, 11231 represents Chick Egg Production, and 112310 represents Chicken Egg Production.

NAICS codes are available in the CRSP database from August 24, 2001 onwards, on a security level. Unknown NAICS codes are blank. For additional information on NAICS codes, please refer to the Executive Office of the President Office of Management and Budget's most current North American Industry Classification System manual, or visit the US Census Bureau's website at http://www.census.gov/epcd/www/naics.html.

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	20010824	Monthly:	20010824	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`snaics/0`
Array	`names`			Member and/or Arrays	`names`		*ts_print* Monthly Usage	`msnaics/0`
Element	`naics`			Element	`naics`		*stk_print* Option(s)	`/xn`[1]

[1] When used with `/fs`.

North American Industry Classification System - Header

North American Industry Classification System - Header is an 6-character code used to group companies with similar products or services. It contains the most current NAICS code in the database. See *North American Industry Classification System* (NAICS) for additional detail on NAICS codes. N.B. this field includes data starting on 20010824.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	20010824	Monthly:	20010824	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`hnaics`			Element	`hnaics`		*stk_print* Option(s)	`/hn`

[1] Executive Office of the President Office of Management and Budget. North American Industry Classification System. Page 27. United States, 1997.

Number of Array Elements

Number of Array Elements is the count of actual event structures available in a CRSP event object for the current entity.

Primary Concept(s)	Base CRSPAccess Data Structure(s);Event Array Objects						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Count	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	CRSP_ARRAY			Type or Subtype	crsp_array		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Array	n/a		*ts_print* Monthly Usage	n/a
Element	num			Element	num		*stk_print* or *ind_print* Option(s)	n/a

Number of Index List Types

Number of Index List Types is the number of lists available for *INDNOs* in an index set. It is set to one in index group and index series sets. If there are no data for a list, *Number of Available Array Elements* for the list is set to zero.

Primary Concept(s)	Index List History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Count	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	ind			Type or Subtype	n/a		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	listtypes			Element	listtypes		*ind_print* Option(s)	n/a

Number of Index Types

Number of Index Types is the number of indices or portfolio time series available for *INDNOs* in an index set. In a series set, the *Number of Index Types* is always 1. In a group set, the *Number of Index Types* is always 17. Not all *INDNOs* have data for all available time series. If there are no data for one of the available time series, *Begin of Valid Data* and *End of Valid Data* of that time series are set to zero.

Primary Concept(s)	Base CRSPAccess Data Structure(s);Index Time Series						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Count	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	n/a			Type or Subtype	n/a		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	indtypes			Element	indtypes		*ind_print* Option(s)	n/a

Number of Periods in Calendar

Number of Periods in Calendar is the number of periods in a CRSP calendar. This is the last calendar period with valid calendar or time series data. In daily, monthly, and weekly calendars, the last calendar period represents the last trading date with available prices. Annual and quarterly calendars are extended to the end of the next calendar year after the last day of prices.

Primary Concept(s)	Base CRSPAccess Data Structure(s); Calendar						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Range	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND
Object	`CRSP_CAL`			Type or Subtype	`crsp_cal`		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Array	n/a		*ts_print* Monthly Usage	n/a
Element	`ndays`			Element	`ndays`		*stk_print* or *ind_print* Option(s)	n/a

Number of Portfolio Types

Number of Portfolio Types is the maximum number of different portfolio methodologies available in a stock set. It is set to 9 in daily databases and 8 in monthly databases. If there are no data for a portfolio time series, *Begin of Valid Data* and *End of Valid Data* are both set to zero.

Primary Concept(s)	Portfolio Statistics and Assignment Arrays, Shares Outstanding Observations Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:		Monthly:		Unit	Count	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK*
Object	n/a			Type or Subtype	n/a		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`porttypes`			Element	`porttypes`		*ind_print* Option(s)	n/a

Number of Rebalancing Types

Number of Rebalancing Types is the number of portfolio rebalancing arrays available for *INDNOs* in an index set. In a series set, *Number of Rebalancing Types* is always 1. In a group set, *Number of Rebalancing Types* is always 10. Not all *INDNOs* have rebalancing data for all available portfolios. If there are no rebalancing data for one of the available rebalancing series, *Number of Array Elements Series* for that array is set to zero.

Primary Concept(s)	Rebalancing History Arrays						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Count	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	n/a			Type or Subtype	n/a		*ts_print* Daily Usage	n/a
Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`rebaltypes`			Element	`rebaltypes`		*ind_print* Option(s)	n/a

Object Array

Object Array identifies the generic array used to store time series, event, or header data in CRSPAccess object data structures.

Primary Concept(s)		Base CRSPAccess Data Structure(s); Time Series Objects, Event Array Objects, Header Objects					Data Type	generic pointer
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Pointer	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND

Object	`CRSP_TIMESERIES`
	`CRSP_ARRAY`
	`CRSP_ROW`
	`CRSP_CAL`
				Type or Subtype	Variable Specific Version of C Usage Objects. FORTRAN-95 does not have generic object arrays.		*ts_print* Daily Usage	n/a

Array	n/a			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`arr`			Element	Variable-specific Array		*stk_print* or *ind_print* Option(s)	n/a

Object Type Code

Object Type Code is an integer code defining the type of object data structure. *Object Type Code*s are assigned as follows:

Structure Name	Object Type Code
`CRSP_CAL`	1
`CRSP_TIMESERIES`	2
`CRSP_ARRAY`	3
`CRSP_ROW`	5

Primary Concept(s)		Base CRSPAccess Data Structure(s); Time Series Objects, Event Array Objects, Header Objects, Calendar Objects					Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK, IND

Object	`CRSP_TIMESERIES`
	`CRSP_ARRAY`
	`CRSP_ROW`
	`CSRP_CAL`
				Type or Subtype	`crsp_ts`
					`crsp_array`
					`crsp_row`
							ts_print Daily Usage	n/a

Array	n/a			Member and/or Arrays	`crsp_cal`		*ts_print* Monthly Usage	n/a
Element	objtype			Element	`objtype`		*stk_print* or *ind_print* Option(s)	n/a

Partition Subset Screening Structure

Partition Subset Screening Structure, like the *Index Subset Screening Structure*, is a set of fields used to restrict a database using various screening variables. The screen fields are *Universal Subset Type Code*, *First Trading Date Allowed in Restriction*, *Index Restriction End Date*, *Valid Exchange Codes in Universe*, *Valid NASDAQ Market Groups in Universe*, *Valid When-Issued Securities in Universe*, *Valid Incorporation of Securities in Universe*, and *Share Code Screen Structure*. *Partition Subset Screening Structure* screens are used to restrict the securities used in defining partition breakpoints of an index.

Primary Concept(s)	Index Header						Data Type	structure
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Set (header information)	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	IND
Object	`indhdr_row`		Type or Subtype	`indhdr`			*ts_print* Daily Usage	n/a
Array	`indhdr`		Member and/or Arrays	`partuniv`			*ts_print* Monthly Usage	n/a
Element	`partuniv`		Element	n/a			*ind_print* Option(s)	`/hr`

Payment Date

Payment Date is the integer date in `YYYYMMDD` format upon which dividend checks are mailed or other distributions are made. It is set to zero if unavailable. For a merger, exchange or total liquidation where the company disappeared *Payment Date* is, by convention, set equal to the date of the last price or *Delisting Date*.

For rights offerings the *Payment Date* is set equal to the record date, found in "Moody's Dividend Record" by convention.

Payment Dates of liquidating payments after delisting are reported when available and are set to 0 when unavailable.

Primary Concept(s)	Distribution Event Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`dists_arr`		Type or Subtype	`dists_arr`			*ts_print* Daily Usage	n/a
Array	`dists[]`		Member and/or Array	`dists()`			*ts_print* Monthly Usage	n/a
Element	`paydt`		Element	`paydt`			*stk_print* Option(s)	`/di`

Permanent Number of Securities in Index List

Permanent Number of Securities in Index List is the CRSP *PERMNO* of a security that is assigned to an index specified with an Index List Array.

Primary Concept(s)	Index List History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	IND
Object	`list_arr[]`		Type or Subtype	`ind_list_arr()`			*ts_print* Daily Usage	n/a
Array	`list[][]`		Member and/or Arrays	`list(,)`			*ts_print* Monthly Usage	n/a
Element	`permno`		Element	`permno`			*ind_print* Option(s)	n/a

PERMCO

PERMCO is a unique permanent company identification number assigned by CRSP to all companies with issues on a CRSP File. This number is permanent for all securities issued by this company regardless of name changes.

Primary Concept(s)	Header Identification and Summary Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	header_row			Type or Subtype	stkhdr		*ts_print* Daily Usage	permco/0
Array	header			Member and/or Array	n/a		*ts_print* Monthly Usage	mpermco/0
Element	permco			Element	permco		*stk_print* Option(s)	/hh, /hr, /hrl, /hn

PERMNO

PERMNO is a unique permanent security identification number assigned by CRSP to each security. Unlike the *CUSIP*, *Ticker Symbol*, and *Company Name*, the PERMNO neither changes during an issue's trading history, nor is it reassigned after an issue ceases trading. The user may track a security through its entire trading history in CRSP's files with one PERMNO, regardless of name or capital structure changes. The Stock Data are sorted and indexed by this field. *PERMNO* is currently a 5-digit integer for all common securities in the CRSP files. The range -999989 to -100 and 100 to 999989 is reserved for CRSP *PERMNO* assignments.

Primary Concept(s)	Header Identification and Summary Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	header_row			Type or Subtype	stkhdr		*ts_print* Daily Usage	permno
Array	header			Member and/or Array	n/a		*ts_print* Monthly Usage	mpermno
Element	permno			Element	permno		*stk_print* or *ind_print* Option(s)	/hh, /hr, /hrl, /hn

Portfolio Assignment Number

Portfolio Assignment Number is the integer assignment of a security for the portfolio type for the time period. If no assignment is made for the security during the period, *Portfolio Assignment Number* is set to zero.

Portfolio assignment rules are based on the index methodology of a portfolio type. See "Chapter 3: CRSP Index Methodologies" on page 27 The time period of *Portfolio Assignment Number* is the time the security is held in the portfolio, but is usually based on the statistic in a previous period.

Primary Concept(s)	Portfolio Statistics and Assignment Time Series						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK‡
Object	port_ts[]			Type or Subtype	port_ts()		*ts_print* Daily Usage	port/#
Array	port[][]			Member and/or Arrays	port()		*ts_print* Monthly Usage	mport/#
Element	port			Element	port		*stk_print* Option(s)	/dy#

Portfolio Building Rules Structure

Portfolio Building Rules Structure is a group of fields describing rules used to build index portfolios. *Portfolio Building Rules Structure* contains fields *Index Basic Rule Type Code*, *Index Function Code for Buy Rules*, *Index Function Code for Sell Rules*, *Index Function Code for Generating Statistics*, and *Index Statistic Grouping Code*.

Primary Concept(s)	Index Header						Data Type	structure
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Set (portfolio building rules)	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`rules`		*ts_print* Monthly Usage	n/a
Element	`rules`			Element	n/a		*ind_print* Option(s)	`/hr`

Portfolio Number if Subset Series

Portfolio Number if Subset Series is the portfolio number within an index group to which this index series belongs. The Index *Primary Link* variable contains the *Permanent Index Group Identification Number.* This index is the n^{th} series within the group index, or zero if it is a stand-alone series.

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`portnum`			Element	`portnum`		*ind_print* Option(s)	`/hh`, `/hr`

Portfolio Number in Associated Index

Portfolio Number in Associated Index is the portfolio number within an associated index group defined in *INDNO of Associated Index*. The associated index breakpoint information for that portfolio is used for this index. It is set to zero if no outside rebalancing information is used to build this index.

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`assign`		*ts_print* Monthly Usage	n/a
Element	`assign.asport`			Element	`asport`		*ind_print* Option(s)	`/hr`

Portfolio Statistic Value

Portfolio Statistic Value is the statistic calculated for the security for the time period. If no statistic is calculated, a missing value dependent on the portfolio statistic is set. Missing market capitalizations are set to zero, and missing beta or standard deviations are set to –99.0.

Statistic calculations are based on the methodology of the portfolio type. The statistic is for the current period, and usually determines the portfolio assignment of the next period.

Primary Concept(s)	Portfolio Statistics and Assignment Time Series						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Mathematical value	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK*
Object	`port_ts[]`			Type or Subtype	`port_ts()`		*ts_print* Daily Usage	`portstat`
Array	`port[][]`			Member and/or Array	`port(,)`		*ts_print* Monthly Usage	`mportstat`
Element	`stat`			Element	`stat`		*stk_print* Option(s)	`/dy`

Price Alternate

Monthly: *Price Alternate* is an alternate monthly price derived from daily prices. *Price Alternate* contains the last non-missing price in the month. The date of this price is stored in the *Price Alternate Date* field. *Price Alternate* is set to zero if no prices are available in the month. New issues that do not begin on the last trading date of a month have the first price and date of the first price at the beginning of the *Price Alternate* and *Price Alternate Date* time series arrays.

Price Alternate is available only on monthly databases during time periods when daily data are available.

Primary Concept(s)	Auxiliary Time Series Data						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	n/a	Monthly:	196207	Unit	USD	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`altprc_ts`			Type or Subtype	`altprc_ts`		*ts_print* Daily Usage	n/a
Array	`altprc[]`			Member and/or Arrays	`altprc_ts()`		*ts_print* Monthly Usage	n/a
Element	n/a			Element	n/a		*stk_print* Option(s)	`/po`

Price Alternate Date

Monthly: *Price Alternate Date* contains the date of the monthly *Price Alternate* (derived from daily data) in `YYYYMMDD` format. If this price is nonzero, then *Price Alternate Date* contains the date of that price. If there are no non-missing prices in the month, then *Price Alternate Date* and *Price or Bid/Ask Average* are set to zero. New issues that do not begin on the last trading date of a month are set to the first price and date available in the month.

Primary Concept(s)	Auxiliary Time Series Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	n/a	Monthly:	196207	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`numtrd_ts`			Type or Subtype	`altprcdt_ts`		*ts_print* Daily Usage	n/a
Array	`numtrd[]`			Member and/or Array	`altprcdt_ts()`		*ts_print* Monthly Usage	n/a
Element	n/a			Element	n/a		*stk_print* Option(s)	`/pn`

Price or Bid/Ask Average

If the closing price is not available for any given period, the number in the price field is replaced with a bid/ask average. Bid/ask averages have dashes placed in front of them. These do not wrongly reflect negative prices, but serve simply to distinguish bid/ask averages from actual closing prices. If neither price nor bid/ask average is available, *Price or Bid/Ask Average* is set to zero.

If the security of a company is included in the composite pricing network, then the closing price while listed on the exchange on a trading date is the last trading price for that day on the exchange on which the security last traded. Similarly, highs, lows, and volumes include trades on all U.S. exchanges on which that security traded. For example, if a stock trades on both the NYSE and the PACX (Pacific Stock Exchange), and the last trade occurs on the PACX, the closing price on that day represents the closing price on the PACX, not the NYSE.

Price data for NASDAQ securities come directly from the NASD with the close of the day. Automated trades and after hour trading are not reflected in the close, high, low, bid, or ask, but are included in the number of trades and volumes. There is no price data from one trading day applied to the next trading day.

All prices are raw prices as they were reported at the time of trading.

Daily: *Price or Bid/Ask Average* is the closing price or the bid/ask average (negative) for a trading day. If the closing price is not available on any given trading day, the number in the price field is a bid/ask average, not an actual closing price.

Daily trading prices for The NASDAQ National Market securities were first reported November 1, 1982. Daily trading prices for The NASDAQ SmallCap Market were first reported June 15, 1992. *Price or Bid/Ask Average* for NASDAQ securities is always a negative bid/ask average before this time.

Monthly: In a monthly database, *Price or Bid/Ask Average* is the price on the last trading date of the month. The price series begins the first month-end after the security begins trading and ends the last complete month of trading. If the closing price is not available on any given end of month trading day, the number in the price field is a bid/ask average, not an actual closing price.

Primary Concept(s)	Price, Volume, and Return Time Series Arrays						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	USD	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`prc_ts`			Type or Subtype	`prc_ts`		*ts_print* Daily Usage	`prc/0`
Array	`prc[]`			Member and/or Array	`prc()`		*ts_print* Monthly Usage	`mprc/0`
Element	n/a			Element	n/a		*stk_print* Option(s)	`/pp, /dd, /dr, /dx, /dsYYYYMMDD|base.amt|`

Price Open

Daily open prices are available for securities traded on NYSE, AMEX, and NASDAQ exchanges beginning June 15, 1992. They represent the first trade after market opens. For NYSE, additional daily open prices are available between December 1925 and June 1962. Open prices are available for NYSE Arca securities beginning March 8, 2006.

Primary Concept(s)	Auxiliary Time-Series Data						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	n/a	Unit	USD	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`openprc_ts`			Type or Subtype	`openprc_ts`		*ts_print* Daily Usage	`openprc/0`
Array	`openprc[]`			Member and/or Array	`openprc()`		*ts_print* Monthly Usage	`n/a`
Element				Element			*stk_print* Option(s)	`/po`

Primary Exchange

Primary Exchange is a one-character code which identifies the primary exchange on which the security trades.

Code	Exchange
N	NYSE
A	AMEX
Q	NASDAQ
X	Other Exchange
R	ARCA

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`primexch/0`
Array	`names[]`			Member and/or Arrays	`names()`		*ts_print* Monthly Usage	`mprimexch/0`
Element	`primexch`			Element	`primexch`		*stk_print* Option(s)	`/xn`[1]

[1] When used with `/fs`.

Primary Exchange - Header

Primary Exchange - Header is a one-character code which identifies the primary exchange on which the security trades. See *Primary Exchange* for a list of the codes.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	`primexch/2`
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	`mprimexch/2`
Element	`hprimexch`			Element	`hprimexch`		*stk_print* Option(s)	`/hn`

Record Date

Record Date is the record date on which the stockholder must be registered as holder of record on the stock transfer records of the company in order to receive a particular distribution directly from the company. This integer date is coded as YYYYMMDD, and set to 0 if unavailable.

For a merger, exchange, or total liquidation in which the company disappeared, *Record Date* is, by convention, set equal to the date of the last price or *Delisting Date*.

Record dates of liquidating payments after delisting are reported when available, and set to 0 when unavailable.

Primary Concept(s)	Distribution Event Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	dists_arr		Type or Subtype	dists_arr			*ts_print* Daily Usage	n/a
Array	dists[]		Member and/or Array	dists()			*ts_print* Monthly Usage	n/a
Element	rcrddt		Element	rcrddt			*stk_print* Option(s)	/di

Related Assignment Information

Related Assignment Information is a group of fields defining the time periods and associated indices used to form portfolios. It primarily defines the rebalancing periods when the portfolio is reformed based on new information. Related Assignment Information contains the fields *Basic Assignment Type Code*, *INDNO of Associated Index*, *Portfolio Number in Associated Index*, *Calendar Identification Number of Rebalancing Calendar*, *Calendar Identification Number of Assignment Calendar*, and *Calendar Identification Number of Calculations Calendar*.

Primary Concept(s)	Index Header						Data Type	structure
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Set (partition assignment)	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	IND
Object	indhdr_row		Type or Subtype	indhdr			*ts_print* Daily Usage	n/a
Array	indhdr		Member and/or Arrays	assign			*ts_print* Monthly Usage	n/a
Element	assign		Element	n/a			*ind_print* Option(s)	/hr

Restriction Begin Date (Partition or Index)

Restriction Begin Date is the first date, in YYYYMMDD format, of data included in a partition universe restriction or an index universe restriction. *Restriction Begin Date* is set to 0 if there is no date restriction.

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	IND
Object	indhdr_row		Type or Subtype	indhdr			*ts_print* Daily Usage	n/a
Array	indhdr		Member and/or Arrays	induniv or partuniv			*ts_print* Monthly Usage	n/a
Element	induniv.begdt or partuniv.begdt		Element	begdt			*ind_print* Option(s)	/hr

Restriction End Date (Partition or Index)

Restriction End Date is the last date, in `YYYYMMDD` format, of data included in a partition universe restriction or an index universe restriction. *Restriction End Date* is set to 0 if there is no date restriction.

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	IND
Object	`indhdr_row`		Type or Subtype	`indhdr`			*ts_print* Daily Usage	n/a
Array	`indhdr`		Member and/or Arrays	`induniv or partuniv`			*ts_print* Monthly Usage	n/a
Element	`induniv.enddt` or `partuniv.enddt`		Element	`enddt`			*ind_print* Option(s)	`/hr`

Return of Delisted Issues Flag

Return of Delisted Issues Flag is a code describing whether delisting returns are applied to securities delisting from the exchange during a rebalancing period of an index. The following codes are used:

Code	Description
0	Unknown or not applicable
1	Delisting return is applied to issues that delist during the period
2	Issues must have price during period on target exchange to be included in index

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	IND
Object	`indhdr_row`		Type or Subtype	`indhdr`			*ts_print* Daily Usage	n/a
Array	`indhdr`		Member and/or Arrays	`flags`			*ts_print* Monthly Usage	n/a
Element	`flags.delretflag`		Element	`delretflag`			*ind_print* Option(s)	`/hr`

Return Without Dividends

Return without Dividends is the capital appreciation of a security. Ordinary dividends and certain other regularly taxable dividends are excluded from the returns calculation. See "Holding Period Total Return" on page 63 for missing values. The formula is the same as for *Holding Period Total Returns* except that ordinary dividends are not included in *d(t)*.

Primary Concept(s)	Optional Time Series Array DataAuxiliary Time Series Data						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Mathematical value	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	STK
Object	`retx_ts`		Type or Subtype	`retx_ts`			*ts_print* Daily Usage	`retx/0`
Array	`retx[]`		Member and/or Array	`retx_ts()`			*ts_print* Monthly Usage	`mretx/0`
Element	n/a		Element	n/a			*ind_print* Option(s)	`/px, /dr`

Security Status

Security Status is a one-character code that describes the status of a security.

Code	Status
W	When Issued
R	Regular Way
E	Ex-Distributed
Q	Bankruptcy
X	Unknown

Primary Concept(s)		Name History Array					Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	names_arr			Type or Subtype	names_arr		*ts_print* Daily Usage	secstat/0
Array	names[]			Member and/or Arrays	names()		*ts_print* Monthly Usage	msecstat/0
Element	secstat			Element	secstat		*stk_print* Option(s)	/xn[1]

[1] When used with /fs.

Security Status - Header

Security Status - Header is a one-character code that describes the status of a security. See *Security Status* above for a list of available codes.

Primary Concept(s)		Header Identification and Summary Data					Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	header_row			Type or Subtype	stkhdr		*ts_print* Daily Usage	secstat/2
Array	header			Member and/or Arrays	n/a		*ts_print* Monthly Usage	msecstat/2
Element	hsecstat			Element	hsecstat		*stk_print* Option(s)	/hn

Share Class

Share Class describes the class of share and is generally blank. Any letter that identifies the class of stock (e.g., "A" for class A common) is contained in the first position of this field. Classes are assigned by the exchange in cooperation with the company.

Primary Concept(s)		Name History Array					Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	names_arr			Type or Subtype	names_arr		*ts_print* Daily Usage	shrcls/0
Array	names[]			Member and/or Array	names()		*ts_print* Monthly Usage	mshrcls/0
Element	shrcls			Element	shrcls		*stk_print* Option(s)	/n

Share Code

Share Code is a 2-digit integer code describing the type of shares traded. The first digit describes the type of security traded.

First Digit - Share Code - Security Type

Code	Definition
1	Ordinary Common Shares
2	Certificates, Americus Trust Components *(Prime, Score, & Units)*
3	ADRs (American Depositary Receipts)
4	SBIs (Shares of Beneficial Interest)
7	Units (Depositary Units, Units of Beneficial Interest, Units of Limited Partnership Interest, Depositary Receipts, etc), Exchange Traded Funds

Note: "Units" (code 7) does not represent combinations of common stock and anything else, such as warrants.

The second digit gives more detailed information about the type of security traded or company.

Second Digit - Share Code - Security Type

Code	Definition
0	Securities which have not been further defined
1	Securities which need not be further defined
2	Companies incorporated outside the US
3	Americus Trust Components *(Prime, Score, & Units)*, Exchange Traded Funds
4	Closed-end funds
5	Closed-end fund companies incorporated outside the US
8	REIT's (Real Estate Investment Trusts)

For example, a Share Code of 14 represents ordinary common shares of a closed-end fund.

Primary Concept(s)	Name History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`shrcd/0`
Array	`names[]`			Member and/or Array	`names()`		*ts_print* Monthly Usage	`mshrcd/0`
Element	`shrcd`			Element	`shrcd`		*stk_print* Option(s)	`/n`

Share Code - Header

Share Code - Header is the last CRSP Share Code in a specific security's Name History Array. See "Share Code" on page 101 for the description of the CRSP share type coding scheme.

Primary Concept(s)	Header Identification and Summary Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	`shrcd/2`
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	`mshrcd/2`
Element	`hshrcd`			Element	`hshrcd`		*stk_print* Option(s)	`/hn`

Share Code Groupings for Subsets (Partition or Index Restriction)

Share Code Groupings for Subsets is an integer code describing the generic share code groupings used in universe subsets describing the valid issues used when partitioning the market or in the actual index. The following codes are used:

Code	Description
0	No share code restriction or not applicable
1	Common stocks excluding ADRs
2	Common stocks excluding ADRs and foreign incorporated companies
3	Common stocks excluding ADR's, foreign incorporated companies, REITS, and closed end funds
4	Common stocks

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`induniv` or `partuniv`		*ts_print* Monthly Usage	n/a
Element	`induniv.shrcd.sccode` or `partuniv.shrcd.sccode`			Element	sccode		*ind_print* Option(s)	`/hr`

Share Code Screen Structure (Partition or Index Restriction)

Share Code Screen Structure contains fields defining groups of CRSP security share codes included in the subset describing the valid issues used when partitioning the market or in the actual index. See "Share Code" on page 101 for details of the 2-digit share codes used by CRSP in the Share Code variable. The fields in the structure are *Share Code Groupings for Subsets*, *Valid First Digit of Share Code*, and *Valid Second Digit of Share Code*.

Primary Concept(s)	Index Header						Data Type	structure
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Set (Id codes)	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`induniv` or `partuniv`		*ts_print* Monthly Usage	n/a
Element	`induniv.shrcd` or `partuniv.shrcd`			Element	`shrcd`		*ind_print* Option(s)	n/a

Share Type

Share Type is a one character code that identifies a security's share type. N.B. this field is not yet populated.

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	n/a	Monthly:	n/a	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`shrtype/0`
Array	`names[]`			Member and/or Arrays	`names()`		*ts_print* Monthly Usage	`mshrtype/0`
Element	`shrtype`			Element	`shrtype`		*stk_print* Option(s)	`/xn`[1]

[1] When used with `/fs`.

Share Type - Header

Share Type - Header is a one character code identifying a security's share type. N.B. this field is not yet populated.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	n/a	Monthly:	n/a	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	header_row			Type or Subtype	stkhdr		*ts_print* Daily Usage	n/a
Array	header			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	hshrtype			Element	hshrtype		*stk_print* Option(s)	/hn

Shares Outstanding Observation Date

Shares Observation Date is a specific date corresponding to a *Shares Outstanding* value. The shares date is either the statement date from a firm's annual or quarterly report, the *Ex-Distribution Date* of a distribution affecting the shares outstanding, or the date of a shares observation taken from another source.

Primary Concept(s)	Shares Outstanding Observations Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	shares_arr			Type or Subtype	shares_arr		*ts_print* Daily Usage	n/a
Array	shares[]			Member and/or Array	shares()		*ts_print* Monthly Usage	n/a
Element	shrsdt			Element	shrsdt		*stk_print* Option(s)	/sh, /sa

Shares Outstanding Observation End Date

Shares Observation End Date is the last effective date of a shares outstanding observation. It is set to the latest date prior to the *Shares Observation Date* of the next observation. The *Shares Observation End Date* of the last observation is set to the *Delisting Date*. If the *Shares Observation Date* is after the *Delisting Date*, then the *Shares Observation End Date* is set to 99999999.

Primary Concept(s)	Shares Outstanding Observations Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	YYYYMMDD date	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	shares_arr			Type or Subtype	shares_arr		*ts_print* Daily Usage	n/a
Array	shares[]			Member and/or Arrays	shares()		*ts_print* Monthly Usage	n/a
Element	shrsenddt			Element	shrsenddt		*stk_print* Option(s)	/sh, /sa

Shares Outstanding

Shares Outstanding is the number of publicly held shares of securities listed on NYSE, AMEX, NASDAQ, and ARCA stock exchanges, recorded in 1000s. This figure represents the actual, undiluted value.

The Shares Outstanding Observations Array contains observation events and cannot be used to directly find the shares outstanding each calendar period. Utility functions and programs are available to map observations to time series to calculate market capitalization. If there is no time series data over a time period, but the security has a *Shares Outstanding* value, the last valid *Shares Outstanding* value has been carried over.

Primary Concept(s)	Shares Outstanding Observations Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Count (in thousands)	Database Format(s)	`shr/0`
C Usage				FORTRAN-95 Usage			Product Type(s)	`mshr/0`
Object	`shares_arr`			Type or Subtype	`shares_arr`		*ts_print* Daily Usage	`/sh, /sa`
Array	`shares[]`			Member and/or Array	`shares()`		*ts_print* Monthly Usage	CRSPAccess
Element	`shrout`			Element	`shrout`		*stk_print* Option(s)	STK

Shares Outstanding Observation Flag

Shares Outstanding Observation Flag is an integer value indicating the source of the shares outstanding observation:

Integer Value	Description
0	Share structure extracted from CRSP data sources
1	Share structure imputed from a split or other distribution
2	Shares observation added on the date of a change in the name history, using the effective shares outstanding on that date implied by another

Primary Concept(s)	Shares Outstanding Observations Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`shares_arr`			Type or Subtype	`shares_arr`		*ts_print* Daily Usage	n/a
Array	`shares[]`			Member and/or Array	`shares()`		*ts_print* Monthly Usage	n/a
Element	`shrflg`			Element	`shrflg`		*stk_print* Option(s)	`/sh, /sa`

Spread Between Bid and Ask

Monthly: *Spread Between Bid and Ask* is the difference between the closing bid and ask quotes for a security. It is available only when *Ask or High Price* and *Bid or Low Price* are available and *Closing Price or Bid/Ask Average* is a bid/ask average. If *Closing Price or Bid/Ask Average* is zero and *Spread between Bid and Ask* is negative, the spread represents a *Bid or Low Price*. If *Closing Price or Bid/Ask Average* is zero and *Spread between Bid and Ask* is positive, *Spread Between Bid and Ask* represents an *Ask or High Price*. It is set to zero if unavailable.

Primary Concept(s)	Auxiliary Time Series Data						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	n/a	Monthly:	1925	Unit	Mathematical value	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`spread_ts`			Type or Subtype	`spread_ts`		*ts_print* Daily Usage	n/a
Array	`spread[]`			Member and/or Array	`spread()`		*ts_print* Monthly Usage	n/a
Element	n/a			Element	n/a		*stk_print* or *ind_print* Option(s)	`/p2`

Standard Industrial Classification (SIC) Code

The *Standard Industrial Classification (SIC) Code* is used to group companies with similar products or services.

The Standard Industrial Classification Manual contains descriptions of categories recognized by the US Government. SIC Code is an integer between 100 and 9999. The first two digits refer to a major group. The first three digits refer to an industry group. All four digits indicate an industry. Missing SIC Codes are set to zero until March, 2000. The NASDAQ stock exchange reports the first three digits of NASDAQ listed companies based on the company reported primary industry, and CRSP has added a fourth digit of zero. SIC codes of NYSE and AMEX companies are reported with four digits based on SEC groupings. Since March 2000 any new SIC Codes assigned may include four significant digits.

The North American Industry Classification System (NAICS) was introduced in 1997, to succeed the SIC codes. See "North American Industry Classification System Code" on page 88.

Primary Concept(s)	Name History Array						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`siccd/0`
Array	`names[]`			Member and/or Array	`names()`		*ts_print* Monthly Usage	`msiccd/0`
Element	`siccd`			Element	`siccd`		*stk_print* Option(s)	`/n`

Standard Industrial Classification (SIC) Code - Header

Standard Industrial Classification (SIC) Code - Header is the last non-zero SIC Code found in a specific security's name structure. The *Standard Industrial Classification (SIC) Code - Header* is zero for companies for which CRSP has no SIC Codes.

Primary Concept(s)	Header Identification and Summary Data						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	n/a
Array	`header`			Member and/or Array	n/a		*ts_print* Monthly Usage	n/a
Element	`hsiccd`			Element	`hsiccd`		*stk_print* Option(s)	`/hh, /hr, /hrl, /hn`

Statistic Average in Period

Statistic Average in Period is the average statistical value in a portfolio at the beginning of a rebalancing period of a market segment index. It is set to zero if missing or unavailable.

Primary Concept(s)	Index Rebalancing History Arrays						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Mathematical value	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`rebal_arr[]`			Type or Subtype	`rebal_arr()`		*ts_print* Daily Usage	n/a
Array	`rebal[][]`			Member and/or Arrays	`rebal(,)`		*ts_print* Monthly Usage	n/a
Element	`avgstat`			Element	`avgstat`		*ind_print* Option(s)	n/a

Statistic Maximum Identifier

Statistic Maximum Identifier is the identifier of the entity in a portfolio with the maximum statistic at the beginning of a rebalancing period. The identifier can be *PERMNO* or *PERMCO* depending on *Index Statistic Grouping Code*. It is set to zero if unavailable.

Primary Concept(s)	Index Rebalancing History Arrays						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`rebal_arr[]`			Type or Subtype	`rebal_arr()`		*ts_print* Daily Usage	n/a
Array	`rebal[][]`			Member and/or Arrays	`rebal(,)`		*ts_print* Monthly Usage	n/a
Element	`maxid`			Element	`maxid`		*ind_print* Option(s)	`/rb#`

Statistic Maximum in Period

Statistic Maximum in Period is a maximum statistic value in the portfolio at the beginning of a rebalancing period. It is set to zero if unavailable.

Primary Concept(s)	Index Rebalancing History Arrays						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Mathematical value	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`rebal_arr[]`			Type or Subtype	`rebal_arr()`		*ts_print* Daily Usage	n/a
Array	`rebal[][]`			Member and/or Arrays	`rebal(,)`		*ts_print* Monthly Usage	n/a
Element	`maxstat`			Element	`maxstat`		*ind_print* Option(s)	`/rb#`

Statistic Median in Period

Statistic Median in Period is the median statistic value in a portfolio at the beginning of a rebalancing period. It is set to zero if unavailable.

Primary Concept(s)	Index Rebalancing History Arrays						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Mathematical value	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`rebal_arr[]`			Type or Subtype	`rebal_arr()`		*ts_print* Daily Usage	n/a
Array	`rebal[][]`			Member and/or Arrays	`rebal(,)`		*ts_print* Monthly Usage	n/a
Element	`medstat`			Element	`medstat`		*ind_print* Option(s)	n/a

Statistic Minimum Identifier

Statistic Minimum Identifier is the identifier of the entity in a portfolio with the minimum statistic at the beginning of a rebalancing period. The identifier can be *PERMNO* or *PERMCO* depending on the *Index Statistic Grouping Code*. It is set to zero if unavailable.

Primary Concept(s)	Index Rebalancing History Arrays						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`rebal_arr[]`			Type or Subtype	`rebal_arr()`		*ts_print* Daily Usage	n/a
Array	`rebal[][]`			Member and/or Arrays	`rebal(,)`		*ts_print* Monthly Usage	n/a
Element	`minid`			Element	`minid`		*ind_print* Option(s)	`/rb#`

Statistic Minimum in Period

Statistic Minimum in Period is the minimum statistic value in the portfolio at the beginning of the rebalancing period. It is set to zero if unavailable.

Primary Concept(s)	Index Rebalancing History Arrays						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Mathematical value	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`rebal_arr[]`			Type or Subtype	`rebal_arr()`		*ts_print* Daily Usage	n/a
Array	`rebal[][]`			Member and/or Arrays	`rebal(,)`		*ts_print* Monthly Usage	n/a
Element	`minstat`			Element	`minstat`		*ind_print* Option(s)	`/rb#`

Sub-Exchange

Sub-Exchange is a one-character code which identifies a subset of the *Primary Exchange* on which the security trades. N.B. this field is not yet populated.

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	n/a	Monthly:	n/a	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`subexch/0`
Array	`names[]`			Member and/or Arrays	`names()`		*ts_print* Monthly Usage	`msubexch/0`
Element	`subexch`			Element	`subexch`		*stk_print* Option(s)	`/xn`[1]

[1] When used with `/fs`.

Sub-Exchange - Header

Sub-Exchange - Header is a one-character code which identifies a sub-set of the *Primary Exchange* on which the security trades. N.B. this field is not yet populated.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	n/a	Monthly:	n/a	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	n/a
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`hsubexch`			Element	`hsubexch`		*stk_print* Option(s)	`/hn`

Ticker Symbol

Ticker Symbol is an alphabetic symbol assigned to each security by an exchange. Tickers can be reused over time. The combination of *Ticker Symbol*, *Share Class*, *Exchange Code*, and *Calendar Trading Date* uniquely identifies a security. A ticker may be one to three characters for NYSE and AMEX securities or four to five characters for NASDAQ securities.

NASDAQ trading tickers have four base characters and may include a fifth character suffix that provides information about an issue's type or temporary information about an issue's status. CRSP only includes the suffix when it provides permanent descriptive information. The following table describes the suffixes appearing on the CRSP file:

NASDAQ Fifth Character Suffixes

Suffix	Definition
A	Class A
B	Class B
S	Shares of Beneficial Interest
U	Unit
V	When-issued
Y	ADR
Z	Miscellaneous common issues

Occasionally NASDAQ will add two additional suffixes to the base ticker to identify certain issues. However, because the NASDAQ ticker field only allows for five characters, one letter of the base ticker will be dropped. For example:

> If a foreign company with a class A stock has a base ticker symbol ABCD, NASDAQ adds two additional characters, A and F. Due to NASDAQ's limited fields, they will delete a letter from the base ticker, so ABCDAF would be truncated to ABCAF.

There is no guarantee that the ticker suffix matches a share type. The *Share Code* variable should be used to determine the security's share type.

NASDAQ tickers before 1982 in an issue's name history are presumed to represent legitimate trading symbols for that issue at some point in time, although these symbols may be listed out of proper chronological sequence. In addition, the NASDAQ file ticker symbols provided do not necessarily constitute a definitive list of all symbols used throughout an issue's trading history. Due to source limitations, the ticker field may be blank in name histories of NASDAQ securities that stopped trading from the early 1970s through the early 1980s.

NYSE tickers prior to July 1962 are blank.

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1962	Monthly:	196207	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`ticker/0`
Array	`names[]`			Member and/or Array	`names()`		*ts_print* Monthly Usage	`mticker/0`
Element	`ticker`			Element	`ticker`		*stk_print* Option(s)	`/n`

Ticker Symbol - Header

Ticker Symbol - Header is set to the first seven characters of *Trading Symbol - Header* for active issues, and is blank for delisted issues. *The Trading Symbol - Header* contains the symbol used by automated trading systems, including all supplied class, share type, or status suffixes, with no punctuation for active securities in the file. There are no current trading symbols longer than seven characters.

Ticker Symbol - Header is used in CRSP access routines and utility programs using ticker as a database key. Usage is unchanged, but input lists built on previous conventions of `TICKER.SHRCLS` for NYSE and AMEX securities must be changed to comply with the new data. This change provides a field in the CRSP database that contains an exact match with symbols available directly from exchanges and other sources.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1962	Monthly:	196207	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	n/a
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`htick`			Element	`htick`		*stk_print* Option(s)	`/hh, /hn`

Trading Denomination

Trading Denomination is a one-character code identifying the tick size of the security. N.B. this field is not yet populated.

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	n/a	Monthly:	n/a	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`denom/0`
Array	`names[]`			Member and/or Arrays	`names()`		*ts_print* Monthly Usage	`mdenom/0`
Element	`denom`			Element	`denom`		*stk_print* Option(s)	`/xn`[1]

[1] When used with `/fs`.

Trading Denomination - Header

Trading Denomination - Header is a one-character code identifying the tick size of the security. N.B. this field is not yet populated.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	n/a
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`hdenom`			Element	`hdenom`		*stk_print* Option(s)	`/hn`

Trading Status

Trading Status is a one-character field containing the trading status of securities.

Code	Status
A	Active
H	Halted
S	Suspended
X	Unknown

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`trdstat/0`
Array	`names[]`			Member and/or Arrays	`names()`		*ts_print* Monthly Usage	`mtrdstat/0`
Element	`trdstat`			Element	`trdstat`		*stk_print* Option(s)	`/xn`[1]

[1] When used with `/fs`.

Trading Status - Header

Trading Status is a one-character field containing the trading status of securities. See *Trading Status* for a list of codes.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	n/a
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`htrdstat`			Element	`htrdstat`		*stk_print* Option(s)	`/hn`

Trading Ticker Symbol

Trading Ticker Symbol is the trading symbol listed by exchanges and consolidated quote systems. It includes all temporary values, share classes, and share type suffixes. There is no punctuation (no periods) in the *Trading Ticker Symbol*. N.B. this field includes data starting on 20020102 for NYSE/AMEX, and 19821101 for NASDAQ.

Primary Concept(s)	Name History Array						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	19821101	Monthly:	198211	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`names_arr`			Type or Subtype	`names_arr`		*ts_print* Daily Usage	`tsymbol/0`
Array	`names[]`			Member and/or Arrays	`names()`		*ts_print* Monthly Usage	`mtsymbol/0`
Element	`tsymbol`			Element	`tsymbol`		*stk_print* Option(s)	`/xn`[1]

[1] When used with `/fs`.

Trading Ticker Symbol - Header

Trading Ticker Symbol - Header is the most current trading symbol on file, listed by exchanges and consolidated quote systems. It includes all temporary values, share classes, and share type suffixes.

Primary Concept(s)	Header Identification and Summary Data						Data Type	character
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	19821101	Monthly:	198211	Unit	Id	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`header_row`			Type or Subtype	`stkhdr`		*ts_print* Daily Usage	n/a
Array	`header`			Member and/or Arrays	n/a		*ts_print* Monthly Usage	n/a
Element	`htsymbol`			Element	`htsymbol`		*stk_print* Option(s)	`/hn`

Unit of Trade

Unit of Trade is the number of shares in a round lot. N.B. this field is not yet populated.

Universe Subset Type Code (Partition or Index Restriction)

Universe Subset Type Code is an integer code defining a set of restrictions used to define the universe of stocks used to create partitions for an index or for the actual index. The following codes are used:

Code	Description
0	Identifier restriction not applicable
10	NYSE common excluding foreign, ADRs, REIT, Closed End Funds
11	NYSE/AMEX common excluding foreign, ADRs, REIT, Closed End Funds
12	NYSE/AMEX/The NASDAQ National Market common excluding foreign, ADRs, REIT, Closed End Funds
20	NYSE common excluding ADRs
21	AMEX common excluding ADRs
22	NYSE/AMEX common excluding ADRs
23	NASDAQ common excluding ADRs
24	NYSE/AMEX/NASDAQ common excluding ADRs
30	NYSE common
31	AMEX common
32	NYSE/AMEX common
33	NASDAQ common
34	NYSE/AMEX/NASDAQ common
35	NYSE common excluding ADRs and foreigns
36	AMEX common excluding ADRs and foreigns
37	NYSE/AMEX common excluding ADRs and foreigns
38	NASDAQ common excluding ADRs and foreigns
39	NYSE/AMEX/NASDAQ common excluding ADRs and foreigns
40	ARCA common excluding ADRs
41	ARCA common
42	NYSE/AMEX/NASDAQ/ARCA common excluding ADRs
43	NYSE/AMEX/NASDAQ/ARCA common

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`partuniv` or `induniv`		*ts_print* Monthly Usage	n/a
Element	`partuniv.univcode` or `induniv.univcode`			Element	univcode		*ind_print* Option(s)	`/hr`

Valid Exchange Codes in Universe (Partition or Index Restriction)

Valid Exchange Codes in Universe is an integer code indicating the base exchanges in the universe used to partition an index or to populate the actual index. The following table lists the base codes used. The sum of two or more codes indicates all selected exchanges are valid.

Code	Description
0	No exchange restriction
1	NYSE
2	AMEX
4	NASDAQ Stock Market
8	ARCA

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`partuniv` or `induniv`		*ts_print* Monthly Usage	n/a
Element	`partuniv.wantexch` or `induniv.wantexch`			Element	wantexch		*ind_print* Option(s)	`/hr`

Valid First Digit of Share Code (Partition or Index Restriction)

Valid First Digit of Share Code is an integer code describing the valid digits in the first digit of the share code in a subset universe used to partition an index or in the actual index. *Valid First Digit of Share Code* is the decimal representation of a 10-digit binary number. The `nth` bit of the binary number is 1 if an `n` in the first digit of the *Share Code* is valid in the subset, and a 0 otherwise.

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`partuniv` or `induniv`		*ts_print* Monthly Usage	n/a
Element	`partuniv.shrcd.fstdig` or `induniv.shrcd.fstdig`			Element	`fstdig`		*ind_print* Option(s)	`/hr`

Valid Incorporation of Securities in Universe (Partition or Index Restriction)

Valid Incorporation of Securities in Universe describes the incorporation of companies selected in a subset universe used to partition an index or in the actual index. The following integer codes are used.

Code	Description
0	Not applicable or no restriction by country of incorporation
1	Companies incorporated outside of the US are excluded

Primary Concept(s)	Index Header	Data Type	integer number

Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1962	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	indhdr_row			Type or Subtype	indhdr		*ts_print* Daily Usage	n/a
Array	indhdr			Member and/or Arrays	partuniv or induniv		*ts_print* Monthly Usage	n/a
Element	partuniv.wantinc or induniv.wantinc			Element	wantinc		*ind_print* Option(s)	/hr

Valid NASDAQ Market Groups in Universe (Partition or Index Restriction)

Valid NASDAQ Market Groups in Universe is an integer code indicating valid NASDAQ markets in the universe subset used to partition an index or used in the actual index. The NASDAQ National Market is a subset of The NASDAQ Stock Market^{SM}. The following codes are used:

Code	Description
0	No National Market restriction, or not applicable
1	Only issues listed on The NASDAQ National Market are included

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	19820402	Monthly:	198204	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	indhdr_row			Type or Subtype	indhdr		*ts_print* Daily Usage	n/a
Array	indhdr			Member and/or Arrays	partuniv or induniv		*ts_print* Monthly Usage	n/a
Element	partuniv.wantnms or induniv.wantnms			Element	wantnms		*ind_print* Option(s)	/hr

Valid Second Digit of Share Code (Partition or Index Restriction)

Valid Second Digit of Share Code is an integer code describing the valid digits in the second digit of the *Share Code* in a subset universe used in an index partition or in the actual index. *Valid Second Digit of Share Code* is the decimal representation of a 10-digit binary number. The `nth` bit of the binary number is 1 if an `n` in the second digit of the *Share Code* is valid in the subset, and a 0 otherwise.

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	indhdr_row			Type or Subtype	indhdr		*ts_print* Daily Usage	n/a
Array	indhdr			Member and/or Arrays	partuniv or induniv		*ts_print* Monthly Usage	n/a
Element	partuniv.shrcd.secdig or induniv.shrcd.secdig			Element	secdig		*ind_print* Option(s)	/hr

Valid When-Issued Securities in Universe (Partition or Index Restriction)

Valid When-Issued Securities in Universe is an integer code describing the types of when-issued trading allowed in a subset universe used in an index partition or in the actual index. The following codes are used:

Code	Description
0	No when-issued restrictions, or not applicable
10	Initial when-issued trading is included when available. Ex-distribution trading is excluded. When-issued trading during reorganizations is included.
110	Initial when-issued trading is excluded until issue attains regular-way status. Ex-distribution trading is excluded. When-issued trading during reorganizations is included.

Primary Concept(s)	Index Header						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Code	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	IND
Object	`indhdr_row`			Type or Subtype	`indhdr`		*ts_print* Daily Usage	n/a
Array	`indhdr`			Member and/or Arrays	`partuniv` or `induniv`		*ts_print* Monthly Usage	n/a
Element	`partuniv.wantwi` or `induniv.wantwi`			Element	`wantwi`		*ind_print* Option(s)	`/hr`

Volume Traded

Volume Traded is the integer raw number of shares traded during the calendar period. It is expressed in units of one share, for daily data, and on hundred shares for monthly data. Our data source for NYSE/AMEX reports the number rounded to the nearest hundred. For example, 12,345 shares traded will be reported on the NASDAQ Stock Exchange as 12,345 and on the NYSE or AMEX exchanges as 12,300. Volume is set to –99 if the value is missing. A volume of zero usually indicates that there were no trades during the time period and is usually paired with bid/ask quotes in price fields.

On NASDAQ, volumes of after-hours trades are included in the current day, while the trades or quotes are included the next day. Therefore, it is possible to have bid/ask or missing price quotes paired with nonzero volumes. Trades on all exchanges connected to NASDAQ's composite pricing network and all late trades are included in the volume. There are no volumes available on NASDAQ prior to November 1, 1982.

Until June 15, 1992, NASDAQ reported volumes differently on the NASDAQ National Market and NASDAQ SmallCap Market. On the National Market, the volume of each transaction was reported by one part involved in the transaction. On the SmallCap Market, all market makers of a security made two volume reports at the end of the market day, the total number of shares they bought and the total number of shares they sold. The NASDAQ system summed the greater figure (whether buy or sell) from the market reports to create daily volume figures.

NYSE/AMEX volumes are the sum of volumes on all U.S. exchanges where that security traded that day.

Daily: *Volume Traded* is the total raw number of shares of a stock traded on that day, and is not adjusted for splits during the month and it does not contain over-allotments.

Monthly: *Volume Traded* is the sum of the trading volumes during that month. Monthly volumes are the sum of shares reported in units of 100, and are not adjusted for splits during the month.

Primary Concept(s)	Price, Volume, and Return Time Series Arrays						Data Type	integer number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Count (monthly in hundreds, daily actual)	Database Format(s)	CRSPAccess
C Usage				FORTRAN-95 Usage			Product Type(s)	STK
Object	`vol_ts`			Type or Subtype	`vol_ts`		*ts_print* Daily Usage	`vol`/0
Array	`vol[]`			Member and/or Array	`vol()`		*ts_print* Monthly Usage	`mvol`/0
Element	n/a			Element	n/a		*stk_print* Option(s)	`/pv`, `/dd`

Weight of Issue

Weight of Issue is the defined weight of an issue within the index during the range indicated in a list defining the index. It is set to zero if weighting is defined based on data and not part of the list definition.

Primary Concept(s)	Index List History Array						Data Type	real number
Date Range Availability and Unit of Item							Database Availability & Utility Usage	
Data Avail.	Daily:	1925	Monthly:	1925	Unit	Mathematical value	Database Format(s)	CRSPAccess
C Usage			FORTRAN-95 Usage				Product Type(s)	IND
Object	`list_arr[]`		Type or Subtype	`list_arr()`			*ts_print* Daily Usage	n/a
Array	`list[][]`		Member and/or Arrays	`list()`			*ts_print* Monthly Usage	n/a
Element	`weight`		Element	`weight`			*ind_print* Option(s)	n/a

CHAPTER 5: CRSP CALCULATIONS

This section contains formulas and methodologies used to derive CRSP variables in the stock and index files and generated by the CRSP data utilities. These are organized alphabetically by name.

Adjusted Data

Price, dividend, shares, and volume data are historically adjusted for split events to make data directly comparable at different times during the history of a security. CRSP provides raw, *Unadjusted Data*, but data utilities *stk_print* and *ts_print* can be used to generate *Adjusted Data*.

An adjustment base date is chosen as the anchor date. All data on this date are unadjusted, and other data are converted based on the split events between the base date and the time of that data. The adjustment base date is usually chosen to be the last available day of trading.

Split events always include stock splits, stock dividends, and other distributions with price factors such as spin-offs, stock distributions, and rights. Shares and volumes are only adjusted using stock splits and stock dividends. Split events are applied on the Ex-Distribution Date.

Price and dividend data are adjusted with the calculation:

$A(t)=P(t)/C(t)$,

where $A(t)$ is the adjusted value at time t, $P(t)$ is the raw value at time t, and $C(t)$ is the cumulative adjustment factor at time t.

Share and volume data are adjusted with the calculation:

$A(t)=P(t)*C(t)$,

where $A(t)$ is the adjusted value at time t, $P(t)$ is the raw value at time t, and $C(t)$ is the cumulative adjustment factor at time t.

In both cases, where C_0 is the adjustment base date, the cumulative adjustment factor is:

if $t=C_0$,$C(t) = 1.0$

if $t>C_0$ and no split events since t-1,$C(t) = C(t-1)$

if $t>C_0$ and a split event with factor f since $t-1$,$C(t) = C(t-1) * f$

if $t>C_0$ and split event change $C(t-1)/f$

if $t<C_0$ and a split event change $C(t+1)*f$

Where factor is typically the *Factor to Adjust Prices* variable + 1.

If there is a gap in trading where possible split events are not known, all adjusted values are set to missing when the gap is between the observation and the adjustment base date.

Monthly: If monthly summary data (*Bid or Low Price* (Page 45), *Ask or High Price* (Page 43), and *Volume Traded* (Page 114)) are adjusted, the adjustment factor cannot take into account adjustments that take place in the middle of the month. Therefore, the result assumes all adjustment events occur on the last trading day of the month. A more accurate monthly adjusted value can be derived by adjusting and resummarizing the underlying daily data.

Annualized Return

Annualized Return is the constant annual return applied to each period in arrays that would result in the actual compounded return over that range. An *Annualized Return* is a special case of a *Geometric Average Return* (Page 120) where the time periods are expressed in terms of years.

Associated Portfolio Returns

Associated Portfolio Returns are a composite of a group of portfolio index series based on a time-dependent portfolio assignment for a security. They are built for each security based on assignments within the specified portfolio type. The associated portfolio return at any time is the return of the portfolio to which the security belongs at that time. If the security is not assigned to a portfolio of that type at the time, the associated portfolio return is set to a missing value.

Capital Appreciation

Capital Appreciation is the change in value of a security over a holding period. It is similar to *Holding Period Total Returns* (Page 63), except that ordinary dividends and certain other regularly taxable dividends are excluded from the returns calculation. The formula is the same as for security *Holding Period Total Returns* except that *d(t)* is usually 0. See *Returns* (Page 122) for missing values. *Capital Appreciation* is also known as *Return without Dividends*.

Compounded Returns

A *Compounded Return* is a measurement of the change of an investment over a time range when individual returns over all subsets of the time range are known. This is equivalent to reinvestment in the investment each time period.

Compounded Returns are calculated using the formula below:

$$r_c = \prod_{i=1}^{n} (1 + r_i) - 1$$

Where

r_c=*Compounded Return*

r_i=return over period `i`. If r_i is missing, that return is ignored and thus in effect treated as a return of 0. If all returns are missing, the result is also missing.

Cumulative Return

A *Cumulative Return* is a compounded return from a fixed starting point. Each period in a time series of *Cumulative Returns* contains the compounded return from the first period in the time series to the end of that period.

Delisting Return

Delisting Return is the return of security after it is delisted. It is calculated by comparing a value after delisting against the price on the security's last trading date. The value after delisting can include a price on another exchange or the total value of distributions to shareholders. If there is no opportunity to trade a stock after delisting before it is declared worthless, the value after delisting is zero. *Delisting Returns* are calculated similarly to total returns except that the value after delisting is used as the current price.

Valid delisting payment information is either a valid price with at least a bid and ask quote within ten trading periods, or a complete set of payments received for the shares. If information after delisting is insufficient to generate a return a missing value is reported.

Monthly: The monthly *Delisting Return* is calculated from the last month ending price to the last daily trading price if no other delisting information is available. In this case the delisting payment date is the same as the delisting date. If the return is calculated from a daily price, it is a partial-month return. The partial-month returns are not truly *Delisting Returns* since they do not represent values after delisting, but allow the researcher to make a more accurate estimate of the *Delisting Returns*.

When valuing a portfolio, the *Delisting Return* or other representation can be used to assign a value to the delisted security. The researcher must decide whether to assign alternate estimated values based on the *Delisting Code (Page 53)* when delisting payment information is unavailable. If using monthly data and an alternate estimate for *Delisting Return* is used, partial month returns should also be adjusted by this factor.

Dividend Amount in Period (ts_print item)

Dividend Amount is the cash adjustment factor in a holding period return time period used to calculate returns. It is an adjusted summation of all distribution cash amounts available in the distribution history with Ex-distribution dates after the previous period and up to and including the current period, adjusted to the basis at the end of the previous period. *Dividend Amount* can be divided into nonordinary and ordinary types. Nonordinary dividends include return of capital distributions. Ordinary dividends are excluded from capital appreciation returns calculations.

To calculate an adjusted Dividend Amount in Period to its basis at the end of a date range, the following formula may be used with data items extracted through ts_print:

Divamt in period adjusted to end of range = divamt / cumfacpr / facpr

Where

divamt = Dividend Amount in Period
cumfacpr = Cumulative Factor to Adjust Prices over a Date Range
facpr = Factor to Adjust Price in Period

Thus, to calculate a total return using adjusted prices and dividends,

Total Return = (adjprc + (divamt / cumfacpr / facpr)) / prev_adjprc -1

Where

adjpr = Price Adjusted, End of Period
prev_adjprc = Price Adjusted, End of Previous Period

Factor to Adjust Prices in Period (ts_print item)

Factor to Adjust Prices in Period is the amount the current price is multiplied by in returns calculations so that current and previous prices are on the same split-adjusted basis. *Factor to Adjust Prices in Period* is derived from the *Factor to Adjust Price* (Page 61) field of distributions with Ex-Distribution Dates after the previous period and up to and including the current period. In simple stock splits, *Factor to Adjust Prices in Period* is distribution *Factor to Adjust Price* plus one.

Excess Returns

An *Excess Return* is defined as the return in excess of a comparable benchmark. The benchmark can be a single associated index series or a composite of a group of portfolio index series based on security and time-dependent portfolio assignments.

If an *Excess Return* is based on a single index series, the *Excess Return* for a period is

$E(t) = R(t) - I(t)$,

where $E(t)$ is the *Excess Return* at time t, $R(t)$ is the security return at time t, and $I(t)$ is the index return at time t. If the security return $R(t)$ is based on a previous price t' that is not the previous time period, $I(t)$ is the compounded index return from $t' + 1$ to t.

If an *Excess Return* is based on associated portfolios, the *Excess Return* for a period is

$E(t) = R(t) - I(p(t),t)$

where $E(t)$ is the *Excess Return* at time t, $R(t)$ is the security return at time t, $p(t)$ is the portfolio assignment of the security at time t, and $I(p(t),t)$ is the return of that portfolio at time t. If the security return $R(t)$ is based on a previous price t' that is not the previous time period, $I(p(t),t)$ is the compounded return of the security's portfolio return from $t' + 1$ to t. If the security is not assigned a portfolio assignment of the given type at time t, $E(t)$ is set to a missing value.

When cumulating *Excess Return*, the security returns and the index returns are cumulated separately before subtracting the difference.

Geometric Average Returns

A *Geometric Average Return* is the constant return applied to each period in a range that would result in the compounded return over that range.

The *Geometric Average Return* is calculated using the formula below:

$$g_n=(1 + r_c)^{1/n} – 1$$

Where

g_n =the *Geometric Average Return* applicable on each subset period n

r_c =the cumulative return over the entire period

n =the number of equal subset periods to average the return

Income Return

Income Return is the return on the ordinary dividends paid to shareholders of a security. It is the ratio of the amount of ordinary dividends since the end of the previous period up to and including the end of the period of interest to the price at the end of the previous period. It is similiar to a dividend yield.

Income Return can be derived from Total Return and Capital Appreciation as follows:

$$iret(t) = tret(t) – aret(t)$$

where *iret* is the income return for time *t*, *tret* is the total return for time *t*, and *aret* is the capital appreciation for time *t*.

See “6.5 Missing Return Codes” on page 134 for missing values.

Index Count

Index Count is the count in an index for a time period is the number of securities in the portfolio during the time period. Rules are based on the specific index or portfolio methodology.

Index Levels

Index Level is the value of an investment relative to its value at one fixed point in time. *Index Levels* allow convenient comparison of the relative performance of the different portfolios or asset classes. Differences arise between the daily *Index Levels* and the *Index Levels* of other frequencies due to compounding; therefore, these series are not directly comparable.

The initial date and value are set arbitrarily, but must be consistent if comparing multiple indices. The *Index Level* for any series at any time after the initial point indicates the value at that time of the initial value invested at the initial point. The *Index Level* of a series is set to zero prior to available data. Let:

i_t=*Index Level* for any series at time t

r_t=return for the period $t - 1$ to t

t_0=the time of the first non-missing return of the series

D_0=initialization date. An arbitrary date where the level is set to the initial value

V_0=initialization value. An arbitrary value the level is set to on the initialization date

then

if $t = D_0$ then $i_t = V_0$

if $t > D_0$ then $i_t = i_{t-1}(1+r_t)$

if $t < D_0$ and $t >= t_0 - 1$ then $i_t = i_{t+1} / (1+r_{t+1})$

if $t < t_0$ then $i_t = 0$

Defined CRSP indices use the following initial dates and levels:

CRSP Stock File Indices are set to 100.00 on December 29, 1972

CRSP Cap-Based Portfolios are set to 1.00 on December 31, 1925

CRSP US Government Treasury and Inflation Indices are set to 100.00 on December 29, 1972

Publicly available indices such as for the S&P 500 Composite and NASDAQ Composite have initial values set by their creators of those indices and do not match the CRSP initializations.

Index Returns

$$R(I) = \frac{\sum_n w_n(I) r_n(I)}{\sum_n w_n(I)}$$

An *Index Return* is the change in value of a portfolio over some holding period. The return on a portfolio *(R(I))* is calculated as the weighted average of the returns for the individual securities in the portfolio: In a value-weighted portfolio, the weight *($w_n(I)$)* assigned to security *n's* return is its total market value $v_n(I)$. CRSP defines the market value of a security *($v_n(I)$)* as the product of its price *($p_n(I - 1)$)* and its number of shares outstanding *($s_n(I - 1)$)*, at the end of the previous trading period.

$$w_n(I) \equiv p_n(I-1)s_n(I-1)$$

In an equally-weighted portfolio, $w_n(I)=1$ for every stock. Such a portfolio would consist of n stocks, with the same dollar amount invested in each stock.

The security returns can be total returns, capital appreciation, or income returns. This determines whether the index is a total return index, a capital appreciation index, or an income return index.

In an index where the individual components are not known, but an index level is available from an external source, such as the Standard & Poor's 500 Composite Index, return is calculated as follows:

r(t) = level(t) / level(t') –1

where t is the current period, t' is the previous period, and the levels are known at the end of the current and previous periods.

The number of shares outstanding for a security on a given day $(s_n^{(I)})$ is derived from the Shares Outstanding Observations Array (Page 104).

Index Weight

The weight of an index for a time period is the total market value of the portfolio at the beginning of the period. The total market value *V(I)* of the portfolio is $V(I) = \sum_n v_n(I) = \sum_n w_n(I)$ where $V_n(I)$ is the market value of one security in the portfolio, and $W_n(I)$ is the weight of that security.

Market Capitalization

Market Capitalization (in 1000s) is a measurement of the size of a security defined as the price multiplied by the number of shares outstanding. CRSP uses the closing price or the absolute value of the bid/ask average from the *Price or Bid/Ask Average* (Page 96) variable and the applicable shares observation from the *Shares Outstanding* Observation Array (Page 104) for each calendar period to calculate *Market Capitalization*.

Rebasing Index Levels

It is possible to rebase an index to make index levels of two index level series comparable if the returns of both indices were created using the same holding periods. To rebase an index, choose a new initial date and value, find the current index level on the new initial date, and multiply the levels on all dates by the new initial value divided by the old initial date index level.

Returns

A *Return* is the change in the total value of an investment in a security over some period of time per dollar of initial investment. Total Return is the *Holding Period Total Return* (Page 63) for a sale of a security on the given day, taking into account and reinvesting all distributions to shareholders. It is based on a purchase on the most recent time previous to this day when the security had a valid price. Usually, this time is the previous calendar period, but may be up to ten calendar periods prior to the calculation.

Returns are calculated as follows:

For time *t* (a holding period), let

t' =time of last available price < *t*

r(t) =return on purchase at *t'*, sale at *t*

p(t) =last sale price or closing bid/ask average at time *t*

d(t) =dividend amount for *t*

f(t) =factor to adjust price in period *t*

p(t') =last sale price or closing bid/ask average at time of last available price < *t*.

$$r(t) = \frac{p(t)f(t) + d(t)}{p(t')} - 1$$

t' is usually one period before *t*, but *t'* can be up to ten periods before *t* if there are no valid prices in the interval. If there is a trading gap with unknown status between *t* and *t'*, the previous price is considered invalid.

In daily databases, dividends are reinvested in the security on the *Ex-Distribution Date* (Page 59). In monthly databases, the returns are holding period returns from month-end to month-end, not compounded daily returns, and dividends are reinvested in the security at month-end.

The *Factor to Adjust Prices in Period* (Page 119) is derived from the distribution history *Factor to Adjust Price* (Page 61) using all distributions with Ex-Distribution dates after the previous period and up to the end of the current period. The dividend amount is derived from the distribution history *Dividend Cash Amount* (Page 57) and *Factor to Adjust Price* in the same range. For example, if a 2-for-1 split is the only distribution event in the time range, *Factor to Adjust Price* is 1.0, *Factor to Adjust Prices in Period* is 2.0, and *Dividend Cash Amount* is 0.0. If a one dollar dividend is the only distribution event in the time range, both *Dividend Cash Amount* and dividend amount are 1.0.

A series of special return codes specify the reason a return is missing:

Missing Value Codes	Code Description
-66.0	Valid current price, but no valid previous price; either first price, unknown exchange between current and previous price, or more than 10 periods between time *t* and the time of the preceding price *t'*
-77.0	Not trading on the current exchange at time *t*
-88.0	No data available to calculate returns.
-99.0	Missing return due to missing price at time *t*

Scholes-Williams Betas

Beta is a statistical measurement of the relationship between two time series, and has been used to compare security data with benchmark data to measure risk in financial data analysis. CRSP provides annual betas computed using the methods developed by Scholes and Williams (Myron Scholes and Joseph Williams, "Estimating Betas from Nonsynchronous Data," *Journal of Financial Economics*, vol 5, 1977, 309-327).

Beta is calculated each year as follows:

$ret_{i,t}$ =log of (1 + return for security i on day t)

$mret_t$ =log of (1 + value-weighted market return on day t)

$mret3_t$ $=mret_{t-1} + mret_t + mret_{t+1}$ (a 3 day moving average market window)

n =number of observations for the year

$$\beta_i = \frac{\sum_t (ret_{i,t}\ mret\ 3_t) - \left(\frac{1}{n}\right)\left(\sum_t ret_{i,t}\right)\left(\sum_t mret\ 3_t\right)}{\sum_t (mret_t\ mret\ 3_t) - \left(\frac{1}{n}\right)\left(\sum_t mret_t\right)\left(\sum_t mret\ 3_t\right)}$$

where summations over *t* are over all days on which security *i* traded, beginning with the first trading day of the year and ending with the last trading day of the year.

There are two portfolio types based on *Scholes-Williams Beta* calculations: NYSE/AMEX and NASDAQ-only.

In the NYSE/AMEX portfolios, only trading prices are considered in the beta calculation, and a security must have traded half the days in a year to be given a non-missing beta for that year. The index used in the calculation is the total returns on the Trade-only NYSE/AMEX Value-Weighted Market Index.

Betas for the NASDAQ portfolios do not use the standard Scholes-Williams trade-only data restriction, since most NASDAQ securities were not required to report transactions until 1992. Removing bid/ask averages would restrict NASDAQ data to only NASDAQ National Market securities after 1982 and NASDAQ SmallCap securities after June 15, 1992. NASDAQ returns based on bid/ask averages have different characteristics from trade-based returns, and betas are provided for comparison. NASDAQ betas are based on the total returns on the NASDAQ Value-Weighted Market Index.

Standard Deviation

Standard Deviation is a statistical measurement of the volatility of a series. CRSP provides annual standard deviations of daily returns using the following calculations:

$ret_{i,t}$ =daily return (trade or average of bid and ask) of security `i` on day *t*.

n =number of observations for the year (of $ret_{i,t}$)

σ_i =yearly standard deviation for the i^{th} company

$$\sigma_i = \sqrt{\frac{\sum_t (ret_{i,t})^2 - \frac{1}{n}\left(\sum_t ret_{i,t}\right)^2}{n-1}}$$

where summation over *t* is over all returns for the i^{th} company in the given calendar year.

A security must have valid returns for eighty percent of the trading days in a year to have a *Standard Deviation* calculated. There are two portfolio types provided by CRSP with annual standard deviations as the statistic, the NYSE/AMEX Standard Deviation Portfolios and the NASDAQ Standard Deviation Portfolios.

Trade-Only Data

CRSP provides *Price or Bid/Ask Average* (Page 96) as the standard daily price field, and derives returns from this field. Bid/ask averages are marked as negative numbers by convention. A trade-only price is derived from *Price or Bid/Ask Average* by setting all bid/ask average prices to missing. Trade-only returns are calculated using trade-only prices. A trade-only index is calculated using trade-only prices and returns.

Unadjusted Data

Unadjusted Data is price, dividend, shares, and volume data reported in the amounts reported at the time of the observations. All CRSP data are provided unadjusted. However, the distribution history can be used to generate *Adjusted Data* (Page 117) from the raw data.

Weighted Return

Weighted Return is the relative weight of a security within a portfolio or index multiplied by its return. In a value-weighted portfolio, *Weighted Return* is the capitalization at the end of the previous period multiplied by the return for the period

CHAPTER 6: CRSP DATA CODING SCHEMES

6.1 Name History Array Codes

Share Type

This table lists the share type codes found in the CRSP stock files. The first digit describes the type of security traded:

Code	Definition
1	Ordinary Common Shares
2	Certificates, Americus Trust Components *(Prime, Score, & Units)*
3	ADRs (American Depositary Receipts)
4	SBIs (Shares Of Beneficial Interest)
7	Units (Depositary Units, Units Of Beneficial Interest, Units Of Limited Partnership Interest, Depositary Receipts, etc.), Exchange Traded Funds
Note: "Units" (code 7) does not represent combinations of common stock and anything else, such as warrants.	

The second digit describes more detailed information about the type of security:

Share Type - Second Digit - Type of Security

Code	Definition
0	Securities Which Have Not Been Further Defined
1	Securities Which Need Not Be Further Defined
2	Companies Incorporated Outside The U.S
3	Americus Trust Components *(Prime, Score, & Units)**, Exchange Traded Funds**
4	Closed-End Funds and Unit Investment Trusts
5	Closed-End Fund Companies Incorporated Outside The US
8	REIT's (Real Estate Investment Trusts)

North American Security Exchange & Indices Codes

The following table is a list of codes for major North American security exchanges and indices found in the CRSP data files:

Code	Exchange Name
-2	Halted by Primary Listing Exchange
-1	Suspended by Primary Listing Exchange
0	Not Trading on Primary Listing Exchange
1	NYSE
2	AMEX
3	NASDAQ
4	ARCA
5	Mutual Funds (As Quoted By NASDAQ)
10	Boston Stock Exchange
13	Chicago Stock Exchange
16	Pacific Stock Exchange
17	Philadelphia Stock Exchange
19	Toronto Stock Exchange
20	Over-The-Counter (Non-NASDAQ Dealer Quotations)
31	When-Issued Trading on NYSE
32	When-Issued Trading on AMEX
33	When-Issued Trading on NASDAQ
34	When-Issued Trading on ARCA

6.2 Distribution Codes

A four-digit code describes distribution events. The first digit describes the distribution in general terms. The second digit describes the form or method of payment. The meaning of the third digit varies with the value of the first digit, and gives a more detailed description of the event. The fourth digit provides information about the tax status of the distribution.

The coding and meanings of the four digits are described below. For digits 2, 3, and 4, special conventions apply: a value of "0" implies that CRSP has not yet discovered the descriptive information for the corresponding digit; a value of "1" implies sources have been checked and the status for the corresponding attribute is actually unspecified, not applicable, or not available for the distribution.

Digit	Code	Meaning
1 Event Type	1	ordinary dividend
	2	liquidating dividend
	3	exchanges and reorganizations
	4	subscription rights
	5	splits and stock dividends
	6	notation of issuance (change in shares outstanding)
	7	general information announcement for dropped issues

Digit	Code	Meaning
2 Payment Method	0	unknown, not yet coded
	1	unspecified or not applicable
	2	cash, United States dollars
	3	cash, foreign currency converted to US dollars
	4	cash, Canadian dollars (now obsolete, converted to US dollars)
	5	same issue of common stock
	6	units including same issue of common stock
	7	an issue of a different common stock which is on the file
	8	other property
3 Dividend Frequency (for first digit = 1 only)	0	unknown, not yet coded
	1	unspecified or not applicable
	2	monthly
	3	quarterly
	4	semi-annual
	5	annual
	6	year-end or final
	7	extra or special
	8	interim
	9	non-recurring
3 Event Descriptor (for first digit = 2 only)	0	unknown, not yet coded
	1	unspecified or not applicable
	3	partial liquidation
	4	step in total liquidation
	5	final liquidation
	6	approval of liquidation
	7	sale of assets resulting in liquidation of company
	8	court proceedings determining status of company assets
3 Event Descriptor (for first digit = 3 only)	0	unknown, not yet coded
	1	unspecified or not applicable
	2	merger
	5	non-ordinary distribution in another stock
	6	reorganization
	7	option of stock
	8	exchange
3 Rights Valuation Method (for first digit = 4 only)	0	unknown, not yet coded
	1	Transferable unknown value (no price or assigned value)
	2	market value of trading right on `exdate`
	3	Fair market value
	4	Value at exdate, calculate
	5	Non-transferable fair market value
	6	Non-transferable value at `exdate`, calculated (based on `recdate` if `exdate` is unavailable)
	7	Non-transferable, unknown value
3 Split Type (for first digit = 5 only)	0	unknown, not yet coded
	1	unspecified or not applicable
	2	split
	3	stock dividend
	4	split & stock dividend
	5	option of cash
	6	distribution of different issue of common; same company
	7	initial distribution of other class of common; same company

Digit	Code	Meaning
3 Types of Offer or Reason for Issuance (for first digit = 6 only)	0	unknown, not yet coded
	1	unspecified or not applicable
	2	step in merger with company on file
	3	step in merger with company not on file
	4	stock conversion
	5	executive option exercise
	6	own tender offer: stock buy-back
	7	own exchange offer: recapitalization
	8	stock offering
3 Event Descriptor (for first digit = 7 only)	0	unknown, not yet coded
	1	bankruptcy filing
	2	negative financial performance
	3	external tender offer results in too few shareholders
	4	internal tender offer results in too few shareholders
	5	US government intervention
	6	foreign or external intervention
	7	company request
	8	failure to meet exchange requirements
4 Tax Status	0	unknown, not yet coded
	1	unspecified or not applicable
	2	normal taxable at same rate as dividends
	3	normal non-taxable
	4	return of capital (i.e., gain recognized, loss not)
	5	gain or loss realized compared with cost
	6	realized capital gain (Investment Companies)
	7	capital gains tax credit
	8	fully taxable as ordinary income to individuals
	9	dividend reinvestment plan qualifies for the limited exclusion provided by Sec. 305(e) of the Internal Revenue Code

Coding convention note for distribution codes with the fourth digit (tax status) coded as 2 or 8: Until 1986, distribution codes 2 and 8 were used in conjunction with one another such that the 2 represented the part of the dividend qualifying for the dividend exclusion and the 8 representing the part that did not. Since the tax reform act of 1986, which eliminated the exclusion, these have been coded as 2's.

The following table describes some of the most commonly coded distribution events in the CRSP stock files. CRSP did not verify the tax status of ordinary dividends in the NYSE/AMEX file after April, 1987 or in the Supplemental NASDAQ file at any time. Instead, CRSP assigned ordinary dividends the default tax code (12*2); that is, US cash dividend, taxable in the normal way as a dividend. If a dividend received is in the form of a security which is traded on the CRSP Stock files, the dividend code will be in the form *7**.

The distribution codes 6***, excepting 6225, are informational. They indicate a significant change in the shares outstanding and the reason for the change. Code 6225 specifies a dividend amount. See the variable DIVAMT for additional information on the 6225 code.

Category	Code	Description
Dividend	1200	US cash dividend, tax status unknown
	1202	US cash dividend, taxable in normal way
	1212	US cash dividend, unspecified frequency, taxable same rate as dividends
	1214	US cash dividend, tax status - return of capital, gain recognized, loss not
	1218	US cash dividend, unspecified frequency, fully taxable as ordinary income to individuals
	1222	US cash dividend, monthly, taxable same rate as dividends
	1224	US cash dividend, monthly, tax status - return of capital, gain recognized, loss not
	1228	US cash dividend, monthly, fully taxable as ordinary income to individuals
	1232	US cash dividend, quarterly, taxable same rate as dividends
	1234	US cash dividend, quarterly, tax status - return of capital, gain recognized, loss not

Category	Code	Description
	1238	US cash dividend, quarterly, fully taxable as ordinary income to individuals
	1239	US cash dividend, quarterly, tax status - dividend reinvestment plan qualifies for the limited exclusion provided by Sec. 305(e) of the Internal Revenue Code
	1242	US cash dividend, semi-annual, taxable same rate as dividends
	1244	US cash dividend, semi-annual, tax status - return of capital, gain recognized, loss not
	1248	US cash dividend, semi-annual, fully taxable as ordinary income to individuals
	1252	US cash dividend, annual, taxable same rate as dividends
	1254	US cash dividend, annual, tax status - return of capital, gain recognized, loss not
	1258	US cash dividend, annual, fully taxable as ordinary income to individuals
	1262	US cash dividend, year-end or final, taxable same rate as dividends
	1272	US cash dividend, extra or special, taxable same rate as dividends
	1274	US cash dividend, extra or special, tax status - return of capital, gain recognized, loss not
	1278	US cash dividend, extra or special, fully taxable as ordinary income to individuals
	1282	US cash dividend, interim, taxable same rate as dividends
	1292	US cash dividend, non-recurring, or proceeds from sale of rights, taxable same rate as dividends
	1312	Cash dividend (foreign currency converted to US), unspecified frequency, tax status - unspecified or not applicable
	1318	Cash dividend (foreign currency converted to US), unspecified frequency, fully taxable as ordinary income to individuals
	1332	Cash dividend (foreign currency converted to US), quarterly, taxable same rate as dividends
	1338	Cash dividend (foreign currency converted to US), quarterly; fully taxable as ordinary income to individuals
	1342	Cash dividend (foreign currency converted to US), semi-annual, taxable same rate dividends
	1348	Cash dividend (foreign currency converted to US), semi-annual, fully taxable as ordinary income to individuals
	1352	Cash dividend (foreign currency converted to US), annual, taxable same rate as dividends
	1372	Cash dividend (foreign currency converted to US), extra or special, taxable same rate as dividends
	1378	Cash dividend (foreign currency converted to US), extra or special, fully taxable as ordinary income to individuals
	1412	Cash dividend return of capital, taxable as normal dividend
	1712	Dividend in other issue on file, unspecified frequency, taxable same rate as dividends
	1713	Dividend in other issue on file, non-taxable
	1714	Dividend in other issue on file, taxable as return of capital
	1718	Dividend in other issue on file, taxable as ordinary income to individuals
	1772	Dividend in other issue on file with an extra or special frequency, taxable same rate as dividends
	1812	Dividend in issue not on file, unspecified frequency, taxable as dividend
	1813	Dividend in issue not on file, non-taxable
	1814	Dividend in issue not on file, taxable as return of capital
	1872	Special Dividend in issue not on file, taxable as normal dividend
	1999*	Missing dividend terms, tax status - dividend reinvestment plan qualifies for the limited exclusion provided by Sec. 305(e) of the Internal Revenue Code
Liquidation	2161	Announcement of liquidation or liquidating plan, tax status unspecified
	2171	Announcement of sale of assets, tax status unspecified
	2181	Liquidation involved in court proceedings, tax status unspecified
	2216	Cash paid in distribution, tax status - realized capital gains, (Investment Companies)
	2234	Cash paid in partial liquidation, tax status - return of capital, gain recognized, loss not
	2235	Cash paid in partial liquidation, tax status - return of capital, gain, loss realized
	2243	Cash paid as a step in liquidation, non-taxable
	2244	Cash paid as a step in liquidation tax status - return of capital, gain recognized, loss not
	2245	Cash paid as a step in liquidation, tax status - return of capital, gain or loss realized
	2255	Cash paid as a final liquidating payment, tax status - return of capital, gain or loss realized
	2744	Other issue on file distributed as a step in liquidation, tax status - return of capital, gain recognized, loss not
	2817	Issue not on file distributed as a step in unspecified liquidation process, tax status - capital gains tax credit
	2844	Issue not on file distributed as a step in liquidation, tax status return of capital, gain recognized, loss not
	2999*	Missing liquidation information, tax status - dividend reinvestment plan qualifies for the limited exclusion provided by Sec. 305(e) of the Internal Revenue Code
Acquisition / Reorganization	3131	Announcement of tender offer - offer not accepted, offer rescinded, or merger failed, tax status unspecified
	3215	Cash received, preferred redeemed, tax status - gain or loss realized compared with cost
	3222	Cash received in a merger, taxable same rate as dividends
	3224	Cash received in a merger, tax status - return of capital gain, gain recognized, loss not
	3225	Cash received in a merger, tax status - gain or loss realized compared with cost
	3285	Cash received in an exchange of stock, tax status - gain or loss realized compared with cost

Category	Code	Description
	3723	Issue of file, received in a non-taxable merger
	3724	Issue on file, received in a merger tax status - return of capital, gain recognized, loss not
	3725	Issue on file, received in a merger, tax status - gain or loss realized compared with cost
	3752	Issue on file, received as a non-ordinary stock distribution, taxable same rate as dividends
	3753	Issue on file, received as a non-ordinary stock distribution, non-taxable
	3762	Issue on file, received as a spin-off in reorganization, taxable same rate as dividends
	3763	Issue on file, received as a spin-off in reorganization, non-taxable
	3764	Issue on file, received as a spin-off in reorganization, tax status - return of capital, gain recognized, loss not
	3783	Issue on file, received as an exchange, non-taxable
	3784	Issue on file, received as an exchange, tax status - return of capital, gain recognized, loss not
	3785	Issue on file, received as an exchange, tax status - gain or loss realized compared with cost
	3823	Issue not on file, received in a merger, non-taxable
	3824	Issue not on file, received in a merger, tax status - return of capital, gain recognized, loss not
	3825	Issue not on file, received in a merger, tax status - gain or loss realized compared with cost
	3852	Issue not on file, received as a non-ordinary distribution in another stock, taxable same rate as dividends
	3853	Issue not on file, received as a non-ordinary distribution, non-taxable
	3854	Issue not on file, received as a non-ordinary distribution, tax status - return of capital, gain recognized, loss not
	3862	Issue not on file, received in a reorganization, taxable as dividend
	3863	Issue not on file, received in a reorganization, non-taxable
	3864	Issue not on file, received in a reorganization, tax status - return of capital, gain recognized, loss not
	3883	Issue not on file, received in an exchange of stock, non-taxable
	3884	Issue not on file, received in an exchange of stock, return of capital (gain recognized, loss not), nontaxable
	3885	Issue not on file, received in an exchange of stock, gain or loss realized compared with cost
	3888*	Partially coded final or other non-ordinary distribution; amount or some terms missing; tax status unknown
	3989*	Debenture without established market value, tax status unknown
Rights	4523	Rights to buy more of this security, at market value, non-taxable
	4533	Rights to buy more of this security at indicated value, non-taxable
	4563	Rights to buy more of this security, non-transferable value at exdate, calculated (based on recdate if exdate unavailable), non-taxable
	4623	Rights to buy 'units' that include this security, non-taxable
	4722	Rights to buy another common issue on file, taxable same rate as dividends
	4822	Rights to buy other securities at market value, taxable same rate as dividends
	4823	Rights to buy other securities, nontaxable
	4833	Rights to buy other securities at indicated value, non-taxable
	4999*	Missing rights distribution, tax status - dividend reinvestment plan qualifies for the limited exclusion provided by Sec. 305(e) of the Internal Revenue Code
Stock	5523	Stock split, non-taxable
	5533	Stock dividend, non-taxable
	5538	Stock dividend, fully taxable as ordinary income to individuals
	5763	Stock distribution in different issue of same company which trades on the file, non-taxable
	5773	Initial stock distribution of other class of common, same company, which is on the file, non-taxable
	5872	Initial stock distribution in different issue of common, same company, which is not on file, taxable same rate as dividends
	5873	Initial stock distribution in different issue of common, same company, which does not trade on the file, non-taxable
Offer/Issuances	6235	Common shares increased by merger with company not on file, tax status - gain or loss realized compared with cost
	6261	Common shares decreased through a company's own tender offer, tax status - unknown
	6511	Common shares increased or decreased for reasons not specified
	6521	Common shares increased by merger with company on file, tax status - unspecified or not applicable
	6531	Common shares increased by merger with company not on file, tax status - unspecified or not applicable
	6541	Common shares increased through stock conversion, tax status - unspecified or not applicable
	6543	Common shares increased through stock conversion, non-taxable
	6561	Common shares reduced through company's buy-back of shares, tax status - unspecified or not applicable
	6571	Common shares increased through company's own exchange offer, tax status - unspecified or not applicable
	6581	Common shares increased through sale of stock other than rights issue, tax status - unspecified or not applicable
	7111	Bankruptcy filing (for any reason) tax status - unspecified or not applicable
	7121	Negative financial performance tax status - unspecified or not applicable
	7131	External tender offer results in too few shareholders tax status - unspecified or not applicable
	7141	Internal tender offer results in too few shareholders tax status - unspecified or not applicable

Category	Code	Description
	7151	US government intervention (SEC intervention, other government intervention or request)
	7161	Foreign or external intervention (non-US government intervention, foreign non-government intervention, "acts of god") tax status - unspecified or not applicable
	7171	Company request (any reason except bankruptcy) tax status - unspecified or not applicable
	7181	Failure to meet exchange requirements tax status - unspecified or not applicable
*This code alerts the user to information that is not coded, and is inconsistent with the conventional distribution-coding scheme.		

6.3 Delisting Codes

Category	Code	Description
Active	100	Issue still trading NYSE/AMEX or NASDAQ.
	150*	Issue still active, but no prices in this version of file.
	160*	Issue stopped trading, but no prices in file after 840831.
	170*	Issue stopped trading, but not delisted from current exchange (suspended or inactive).
Mergers	200	Issue acquired in merger, payment details unknown.
	201	Merged into or in order to form an issue trading on NYSE.
	202	Merged into or in order to form an issue trading on AMEX.
	203	Merged into or in order to form an issue trading on NASDAQ.
	204	Merged into or in order to form an issue trading on ARCA.
	205	When merged, shareholders primarily receive shares of mutual funds.
	231	When merged, shareholders primarily receive common stock or ADRs. Replaces codes 201, 202 and 203. Codes 201-203 are no longer assigned.
	232	When merged, shareholders primarily receive common stock or ADRs. (Merged stock is not maintained on the CRSP file.) Replaces codes 210-220. Codes 210-220 are no longer assigned.
	233	When merged, shareholders receive cash payments.
	234	When merged, shareholders primarily receive preferred stock, bundled units, warrants, or rights, or debentures, or notes, or bundled units.
	235	When merged, shareholders primarily receive other property.
	240*	Flags merger with missing final distribution information.
	241	When merged, shareholders primarily receive common stock and cash, issue on CRSP file.
	242	When merged, shareholders primarily receive common stock and preferred stock or warrants or rights or debentures or notes, issue on CRSP file.
	243	When merged, shareholders primarily receive common stock, issue on CRSP file and other property, issue on CRSP file.
	244	When merged, shareholders primarily receive common stock or ADR, and cash and preferred stock or warrants or rights or debentures or notes. Issue on CRSP file.
	251	When merged, shareholders primarily receive common stock or ADRs and cash. (Merged stock is not maintained on the CRSP file.)
	252	When merged, shareholders primarily receive common stock or ADRs and preferred stock, or warrants, or rights, or debentures, or notes.
	253	When merged, shareholders primarily receive common stock or ADRs and other property.
	261	When merged, shareholders primarily receive cash and preferred stock, or warrants, or rights, or debentures, or notes.
	262	When merged, shareholders primarily receive cash and other property.
	271	When merged, shareholders primarily receive preferred stock or warrants, or rights, or debentures, or notes and other property.
	280	Issue delisted due to merger attempt, but merger attempt failed.
	290	Flags a merger with missing final distribution information. Replaces code 240. Code 240 is no longer assigned.

Category	Code	Description
Exchanges	300	Issue acquired by exchange of stock, details unknown.
	301	Issue exchanged for issue trading on NYSE.
	302	Issue exchanged for issue trading on AMEX.
	303	Issue exchanged for issue trading on NASDAQ.
	304	Issue exchanged for issue trading on ARCA.
	320	Issue exchanged for stock trading Over-the-Counter.
	331	Issue exchanged, primarily for another class of common stock. Replaces codes 301, 302, and 303. Codes 301-303 are no longer assigned.
	332	Issue exchanged, primarily for another class of common stock. (Other stock is not maintained on the CRSP file.)
	333	Issue exchanged, primarily for cash.
	334	Issue exchanged, primarily for preferred stock, or rights, or warrants, or debentures, or notes.
	335	Issue exchanged, primarily for other property.
	340*	Flags an exchange with missing final distribution information.
	341	Flags an exchange, shareholders receive common stock and cash. Issue on CRSP file.
	342	Flags an exchange, shareholders receive common stock and preferred stock or warrants or rights or debentures or notes. Issue on CRSP file.
	343	Flags an exchange, shareholders receive common stock and other property. Issue on CRSP file.
	350*	Flags an exchange attempt that was not sufficient to "kill" issue.
	351	Flags an exchange, shareholders receive common stock and cash. Issue not on CRSP file.
	352	Flags an exchange, shareholders receive common stock and preferred stock, or warrants, or rights, or debentures, or notes. Issue not on CRSP file.
	353	Flags an exchange, shareholders receive common stock and other property. Issue not on CRSP file.
	361	When exchanged, sharesholders primarily receive cash and preferred stock or warrants or rights or debentures or notes.
	362	When exchanged, shareholders primarily receive cash and other property.
	371	When exchanged, shareholders primarily receive preferred stock or warrants or rights or debentures or notes and other property.
	390*	Flags an unsuccessful exchange attempt with missing distribution information.
Liquidations	400	Issue stopped trading as result of company liquidation.
	450	Issue liquidated, final distribution verified, issue closed to further research.
	460	Issue liquidated, no final distribution is verified, issue closed to further research.
	470	Issue liquidated, no final distribution is verified, issue pending further research.
	480	Issue liquidated, no distribution information is available, issue is pending further research.
	490	Issue liquidated, no distributions are to be paid, issue closed to further research.

Category	Code	Description
Dropped	500	Issue stopped trading on exchange - reason unavailable.
	501	Issue stopped trading current exchange - to NYSE.
	502	Issue stopped trading current exchange - to AMEX.
	503	Issue stopped trading current exchange - to NASDAQ.
	504	Issue stopped trading current exchange - to NYSE Arca.
	505	Issue stopped trading current exchange - to Mutual Funds.
	510	Issue stopped trading current exchange - to Boston Exchange.
	513	Issue stopped trading current exchange - to Midwest Exchange.
	514	Issue stopped trading current exchange - to Montreal Exchange.
	516	Issue stopped trading current exchange - to Pacific Stock Exchange.
	517	Issue stopped trading current exchange - to Philadelphia Stock Exchange.
	519	Issue stopped trading current exchange - to Toronto Stock Exchange.
	520	Issue stopped trading current exchange - trading Over-the-Counter.
	550	Delisted by current exchange - insufficient number of market makers.
	551	Delisted by current exchange - insufficient number of shareholders.
	552	Delisted by current exchange - price fell below acceptable level.
	560	Delisted by current exchange - insufficient capital, surplus, and/or equity.
	561	Delisted by current exchange - insufficient (or non-compliance with rules of) float or assets.
	570	Delisted by current exchange - company request (no reason given).
	572*	Delisted by current exchange - company request, liquidation.
	573	Delisted by current exchange - company request, deregistration (gone private).
	574	Delisted by current exchange - bankruptcy, declared insolvent.
	575	Delisted by current exchange - company request, offer rescinded, issue withdrawn by underwriter.
	580	Delisted by current exchange - delinquent in filing, non-payment of fees.
	581	Delisted by current exchange - failure to register under 12G of Securities Exchange Act.
	582	Delisted by current exchange - failure to meet exception or equity requirements.
	583	Delisted by current exchange - denied temporary exception requirement.
	584	Delisted by current exchange - does not meet exchange's financial guidelines for continued listing.
	585	Delisted by current exchange - protection of investors and the public interest.
	586	Delisted by current exchange - composition of unit is not acceptable.
	587	Delisted by current exchange - corporate governance violation.
	588	Conversion of a closed-end investment company to an open-end investment company.
	589	Delisted by current exchange - unlisted trading privileges
	591	Delisted by current exchange - delist required by Securities Exchange Commission (SEC)
Expirations	600	Expired warrant or right
	601	Warrants, rights, preferreds, or units called for redemption
	610	Unit split into its component parts
Domestics that became Foreign	900	A domestic Security becomes foreign
	901	A domestic Security becomes foreign, but continues to trade on NYSE
	902	A domestic Security becomes foreign, but continues to trade on AMEX
	903	A domestic Security becomes foreign, but continues to trade on NASDAQ
	904	A domestic Security becomes foreign, but continues to trade on ARCA
*Discontinued. Replaced with specific codes in the 400-range.		

6.4 NASDAQ Information Codes

TRTSCD	Description
0	unknown
1	active
2	only one market maker
3	suspended
4	inactive
5	delisted

NMSIND	Description
0	unknown
1	SmallCap before June 15, 1992
2	National Market
3	SmallCap after June 15, 1992
4	Capital Market (formerly SmallCap) after July 1, 2006
5	Global Market (formerly National Market) after July 1, 2006
6	Global Select Market - new subset of Global Market after July 1, 2006

NASD Index Code

Code	Description
0	unknown or unavailable.
1	no index.
2	industrial company.
3	bank.
4	other financial institution.
5	insurance company.
6	transportation company.
7	utility company.

6.5 Missing Return Codes

Missing Return Codes

Parameter	RET(I)	Reason For Missing Return
RMISSN	-44.0	missing excess return due to no portfolio assignment.
RMISSD	-55.0	missing delisting return.
RMISSG	-66.0	more than 10 trading days between this day and the day of latest preceding price.
RMISSE	-77.0	not trading on an included exchange for this file.
RMISSR	-88.0	No data available to calculate returns.
RMISSP	-99.0	missing return due to missing price.

APPENDIX A: CRSP TERMINOLOGY

Absolute Time

Absolute time synchronizes data based on actual calendar time. Time periods are specified as the actual calendar dates when the observations occurred, unlike event time where time periods are relative to specified events.

Associated Index

An associated index is a single predefined index series chosen as a benchmark when comparing performance results of a security, user portfolio, or other index. Examples of associated indices are the S&P 500 Composite or the CRSP Cap-Based 9-10 Portfolio on the NYSE/AMEX/NASDAQ National Market.

Associated Portfolio Type

An associated portfolio type is a predefined index group of portfolios chosen as a benchmark when comparing performance results of a security, user portfolio, or other index. When an associated portfolio type is used as a benchmark, the security portfolio assignment for the portfolio is used to determine which portfolio series to use as a benchmark at each point in time. An example of associated portfolios is the CRSP Cap-Based Portfolios on the NYSE/AMEX/NASDAQ National Market.

Binary Code

A binary code is a numeric flag based on the bitwise "or" of several possible values. It is also known as a bit flag or mask. A flag that is the sum of multiple pieces is valid for all those pieces.

Calendars

A CRSP calendar is a set of time periods with header information about those time periods. The calendar time periods are chosen as points of interest rather than all calendar days, and therefore a daily calendar contains only the dates where trading was conducted on a major US exchange. Data are never provided on weekends or trading holidays. The standard identification of a time period is the date, an integer in YYYYMMDD format, at the end of the period.

There are currently five calendars provided with CRSPAccess databases: daily, monthly, weekly, quarterly, and annual. The daily calendar is used to derive the others so that the last trading date in each month, week, quarter, or year is used to build those respective calendars.

Time series data are always associated with one of these calendars. The list of time series observations is synchronized with a calendar so that the n^{th} time series observation is associated with the n^{th} calendar period.

Calendar/Indices

In the CRSPAccess FORTRAN sample programming usage one calendar is available and is combined with selected market index data in a Calendar/Indices structure. The calendar periods are the first field in the file, followed by results and statistics of a value-weighted index, an equal-weighted index, and a composite index. In this format the calendar/indices calendar is applicable to all time series in the corresponding stock file except portfolios.

Calendar Mapping

Calendar mapping is the operation of converting data to be reported at the frequency of a given calendar. The result data reported for each target calendar period depend on the type of input data and conventions used. Header and event data can be mapped by finding the applicable event or making a composite value from all events relevant to the calendar period. Time series data can also be mapped to different calendars by summarizing observations in a time series with a more frequent calendar, or by reporting an average value in a time series with a less frequent calendar.

Capital Appreciation

Capital appreciation is another name for Returns without Dividends. It is the return for a security or index with ordinary dividends excluded.

Corporate Actions

Corporate actions is another name for Dividend and Delisting information.

CRSPAccess Format

CRSPAccess is the name given to the format of CRSP stock and indices data files introduced in 1996. The CRSPAccess format is a binary format with data utilities and C and FORTRAN random access data libraries supported on multiple platforms. The data specifications of the data are not available; access to the data must use the data utilities or application program interfaces provided. A CRSPAccess database contains a set of binary computer files in a single directory and is identified by the path of this directory.

CRSPDB

CRSPDB is another name for a database in CRSPAccess format.

Data Object

A data object is a set of data organized in a common structure. Data in CRSP files use four basic data objects. These are header data, event arrays, time series, and calendars.

Decile Index

A Decile Index is a Market Segment Index with the market divided into ten portfolios each period.

Dividend Yield

Dividend Yield is another name for Income Return. It is the ratio of the ordinary dividends of a security or index to the previous price.

Entity

In CRSP documentation and utility programs, an entity refers to a single security, portfolio, or index. CRSP databases are organized by entity. Stock databases are organized by security, and Indices databases are organized by index.

Equal-Weighted Portfolio

In an equal-weighted portfolio or index, the same amount is invested in all securities each period. A daily equal-weighted portfolio is reweighted each day, while a monthly equal-weighted portfolio is reweighted each month. Therefore compounded daily index returns over a month are not equivalent to monthly index returns for an equal-weighted portfolio.

An equal-weighted portfolio incurs large transaction costs in practice, especially if maintained daily, since shares must be bought and sold each period as prices change to maintain weighting.

Event Arrays

An event array contains a list of unscheduled transactions, or observation or status changes. There is a count of the number of events. The fields in each event are dependent on the specific data item. The time of the event and relevant information are stored for each observation. Unscheduled transactions record events as they occur. The status observations and changes usually contain information that is in effect until modified by another similar event.

Examples of event arrays in CRSP Stock files are the name history and the distribution history. The name history has a new name observation recorded when any name information changes. The distribution history records information about all distributions made to shareholders of a security. CRSP provides utilities and programming tools to retrieve event data, or to convert relevant event data into time series.

Event Study

An event study synchronizes the time series history of securities relative to a selected event in order to measure the effects of that event.

Event Time

Event time is the time relative to a defined event. Each event contains a security and event date supplied by a user based on CRSP data or outside information. Event time is the number of calendar periods in a time series before or after the event.

Exchange Screening

Exchange screening is used to restrict data to issues listed on one or more specified exchanges.

Fractile Index

A fractile index is a general name for a Market Segment Index covering segmentation of the market into a selected number of portfolios.

Header Data

Header data is information relevant to the entire history of an entity. It usually includes identifiers, summary data, and data ranges. All entities have some kind of header data. The components of the header are specific to the header type.

The CRSP stock file header data include the PERMNO, the header CUSIP, and date ranges of a security. CRSP indices file header data include index identifier and flags describing the index methodology.

Index Groups

Index group refers to a set of related index series, where each member in the set is one portfolio made up of different subsets of the total universe. The set of all series in a portfolio index is an index group. Index groups are convenient for example, when comparing a security whose membership may fluctuate from portfolio to portfolio over time.

Index groups are available only in the CRSP US Indices Database and Security Portfolio Assignment Module.

Index Reweighting

In a portfolio, the weight of each entity is based on the rules of that portfolio. The weight indicates the relative holdings of the security within the portfolio. Reweighting refers to the rules for changing the weights of the existing portfolio components over time.

Index Series

An index series refers to data and results of a single portfolio of securities. A single market index, a standard selection of securities, or one decile from a set of decile portfolio indices is an index series. For example, Portfolio 10 of the CRSP Cap-Based Portfolios for NYSE is an index series, whereas all the portfolios of the CRSP Cap-Based Portfolios for NYSE comprise an index group.

INDNO

In CRSPAccess databases, all indices are assigned an `INDNO`. See “Chapter 3: CRSP Index Methodologies” on page 27 for a full list of available CRSP indices and information on the composition and methodologies for calculating the different indices.

Market Index

A market index is a portfolio of all eligible issues in the market, where the market is defined using constant universe restrictions each period based on some identification or data restriction.

Market Segment Index

A market segment index is a type of index where the market of eligible issues is divided into a fixed number of portfolios at different rebalancing intervals based on some rule or statistic. The breakpoint function is continuous so that

all eligible issues are in exactly one portfolio during each period. The partitioning rules and index calculations are dependent on the specific index methodology. The partitions can be used to define composite portfolios combining the membership of portfolios.

A market segment index is also commonly called a Decile Index if there are ten portfolios, and can also be called a Fractile Index in the general case when there are more or less than ten portfolios.

NASDAQ National Market Screening

NASDAQ National Market screening is used to further restrict NASDAQ data based on The NASDAQ Stock Market listings of The NASDAQ National Market and the NASDAQ SmallCap Market.

Non-Ordinary Distributions

Non-ordinary distributions are distributions made to shareholders of a security representing the return of capital. These are factored into the capital appreciation of the security. Any criteria not included in the Ordinary Distribution description (Page 138) are considered non-ordinary.

Ordinary Distributions

Ordinary distributions are distributions made to shareholders of a security from company profits. These represent income to the shareholder and are not included in the capital appreciation of the security.

CRSP returns calculations use the distribution codes to determine whether a distribution is ordinary or non-ordinary. A distribution is considered ordinary if any of the following conditions are true:

- First digit of the Distribution Code is 1 (1***, where *=anything)
- First digit of the Distribution Code is 2, third digit is anything but 3, and fourth digit is 2 or 8 (2*‡2, 2*‡8 where *=anything and ‡=anything except 3)
- Any Distribution Code with a Factor to Adjust Price of either 0 or -1
- First digit of the Distribution Code is 6 and the fourth digit is 2 or 8 (6**2, 6**8 where *=anything)

PERMCO

The CRSP PERMCO is an integer used to uniquely identify each company in a CRSP stock database. The PERMCO does not change historically if the company changes names. All issues of a company have the same PERMCO, so it can be used to find all issues of a company. The PERMCO can be used as a link to another company when stock of a specific issue is not directly involved. An issue cannot belong to more than one company in its history.

As a unique identifier, PERMCO is analogous to the CUSIP issuer number, the first six characters of the CUSIP, except that it does not change if a company changes names.

PERMNO

The PERMNO is an integer used to uniquely identify each security in a CRSP stock database. The PERMNO does not change historically if the security changes name or makes capital changes, and it can be used to track security history or follow data links to other securities.

CRSP follows a security from the point of view of the shareholders. If a security leaves a major exchange and is readmitted representing the same shareholders it is considered the same security. If shareholders exchange their shares, the issue is closed, and a reissue to new shareholders even by the same company is considered a new issue and a new PERMNO is assigned. CRSP always continues a security if the CUSIP does not change, so therefore no CUSIP is present in the history of more than one PERMNO. If there is a merger, CRSP chooses one of the issues as the survivor and continues that history under the same PERMNO. The survivor is typically the issue that represents the largest capitalization in the merged company, but other data sources are consulted if it is nearly an equal partnership.

Portfolio Rebalancing

Rebalancing refers to the act of reforming a portfolio according to rules in the portfolio methodology. This periodic event involves reapplying the rules of the portfolio to buy, sell, or keep issues in the portfolio.

Portfolio Type

Portfolio type is an integer identifying a specific defined CRSP portfolio methodology. Each security has a time series of statistics and portfolio assignments for every available defined portfolio type. There is a corresponding index calculated for each portfolio within a portfolio type. This group of indices can be used to calculate excess returns for individual securities as portfolio assignments change over time.

The portfolio type identifier must be specified to retrieve portfolio data items in CRSPAccess data utilities or programs.

The following table describes the possible portfolio types available for daily and monthly data:

Portfolio Type Description	Rebalancing Calendar	Permanent Index Identification Number	Daily Portfolio Type	Monthly Portfolio Type	Product Availability
NYSE/AMEX/NASDAQ Capitalization Deciles	Annual	1000092	1	1	DA, MA
NYSE/AMEX Capitalization Deciles	Annual	1000052	2	2	IX
NASDAQ Capitalization Deciles	Annual	1000072	3	3	IX
NYSE Capitalization Deciles	Annual	1000012	4	4	IX
AMEX Capitalization Deciles	Annual	1000032	5	5	IX
NYSE/AMEX Beta Deciles	Annual	1000112	6	-	IX
NYSE/AMEX Standard Deviation Deciles	Annual	1000132	7	-	IX
NASDAQ Beta Deciles	Annual	1000152	8	-	IX
NASDAQ Standard Deviation Deciles	Annual	1000172	9	-	IX
Cap-Based NYSE/AMEX/NASDAQ National Market Portfolios	Quarterly	1000357	-	6	IX
Cap-Based NYSE Portfolios	Quarterly	1000317	-	7	IX
Cap-Based NYSE/AMEX Portfolios	Quarterly	1000337	-	8	IX

Selected Index

A selected index is an index where the universe of eligible issues is supplied by an outside source, with given issues or companies and the ranges of membership for each. Selection criteria are dependent on the specific index methodology.

Set Identifier

Set identifier is a predefined subset of a set type in a CRSPAccess database. Data for two different set identifiers with the same set type use the same data variables, but have different characteristics within those structures. For example, daily and monthly stock sets have different set identifiers since the time series are associated with different calendars and different available portfolio types. Multiple set identifiers of the same set type can be present in one CRSPAccess database.

The predefined set identifiers in CRSPAccess stock and indices files are:

Data	Set Type	Set Identifiers
CRSP Stock Data	STK	10 Daily 20 Monthly
CRSP Indices Data	IND	400 Monthly Index Groups 420 Monthly Index Series 440 Daily Index Groups 460 Daily Index Series

Set Type

Set type is a predefined type of financial data supported in a CRSPAccess database. Stock databases support stock (`STK`), index (`IND`), and calendar (`CAL`) set types. Data for each set type have the same data variables, identifiers, and CRSP programming library access functions.

Share Code Screening

Share code screening is used to restrict data to issues with selected share type characteristics. The share types are based on the CRSP share code. Restrictions can be based on the primary type or secondary type of share and company classification included in the share code.

Stock Subset

A stock subset is the selected history of data that meets criteria based on time range and identifying information. CRSP allows subsetting by date range, exchange code, share type, NASDAQ National Market status, and when-issued status. Data in the history that do not meet the selected restrictions are excluded before any extractions or calculations are done.

Time Series

A time series is a list of observations synchronized with a specific calendar of time periods. There is a beginning and ending of valid data and a link to a calendar defining the time periods. Each entity has exactly one observation for each period within its valid range. The observations can be simple values or contain multiple components, depending on the time series. The values of different variables are generated by checking a value at a consistent time in each time period or by summarizing events occurring during the time period.

The primary data variables in the CRSP files such as prices and returns are time series. In daily databases there are observations for each trading day, and in monthly databases there are observations for each month. In both databases there are portfolio time series with statistics and assignments generated when portfolios are rebalanced, usually annually.

Components of a time series include the type of data, the array of data, the associated calendar, and the beginning and ending range of data for an entity. It is possible to manipulate event and header data into time series, or to convert between different time series items, or to convert between different frequencies of observations. CRSP provides utilities and programming options to manipulate the CRSP variables into a wide range of time series data types.

Value-Weighted Portfolio

In a value-weighted portfolio or index, securities are weighted by their market capitalization. Each period the holdings of each security are adjusted so that the value invested in a security relative to the value invested in the portfolio is the same proportion as the market capitalization of the security relative to the total portfolio market capitalization.

APPENDIX B: CUSIP COPYRIGHT INFORMATION

Subscriber agrees and acknowledges that the CUSIP Database is and shall remain valuable intellectual property owned by, or licensed to, CUSIP Service Bureau, Standard & Poor's ("CSB") and the American Bankers Association ("ABA"), and that no proprietary rights are being transferred to Subscriber in such materials or in any of the information contained therein. Subscriber agrees that misappropriation or misuse of such materials will cause serious damage to CSB and ABA and that in such event money damages may not constitute sufficient compensation to CSB and ABA; consequently, Subscriber agrees that in the even of any misappropriation or misuse, CSB and ABA shall have the right to obtain injunctive relief.

Subscriber agrees that Subscriber shall not publish or distribute in any medium the CUSIP standard numbers. Subscriber further agrees that the use of CUSIP numbers is not intended to create or maintain, and does not serve the purpose of the creation or maintenance of, a file of CUSIP numbers for any other third party recipient of such service and is not intended to create and does not serve in any way as a substitute of the CUSIP MASTER TAPE, PRINT, ELECTRONIC and/or CD-ROM Services.

NEITHER CSB, ABA NOR ANY OF THEIR AFFILIATES MAKE ANY WARRANTIES, EXPRESS OR IMPLIED, AS TO THE ACCURACY, ADEQUACY OR COMPLETENESS OF ANY OF THE INFORMATION CONTAINED IN THE CUSIP DATABASE. ALL SUCH MATERIALS ARE PROVIDED TO SUBSCRIBER ON AN "AS IS" BASIS, WITHOUT ANY WARRANTIES AS TO MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE OR USE NOR WITH RESPECT TO THE RESULTS WHICH MAY BE OBTAINED FROM THE USE OF SUCH MATERIALS, NEITHER CSB, ABA NOR THEIR AFFILIATES SHALL HAVE ANY RESPONSIBILITY OR LIABILITY FOR ANY ERRORS OR OMISSIONS NOR SHALL THEY BE LIABLE FOR ANY DAMAGES, WHETHER DIRECT OR INDIRECT, SPECIAL OR CONSEQUENTIAL EVEN IF THEY HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN NO EVENT SHALL THE LIABILITY OF CSB, ABA OR ANY OF THEIR AFFILIATES PURSUANT TO ANY CAUSE OF ACTION, WHETHER IN CONTRACT, TORT, OR OTHERWISE EXCEED THE FEE PAID BY SUBSCRIBER FOR ACCESS TO SUCH MATERIALS IN THE MONTH IN WHICH SUCH CAUSE OF ACTION IS ALLEGED TO HAVE ARISEN. FURTHERMORE, CSB AND ABA SHALL HAVE NO RESPONSIBILITY OR LIABILITY FOR DELAYS OR FAILURES DUE TO CIRCUMSTANCES BEYOND ITS CONTROL.

Subscriber agrees that the foregoing terms and conditions shall survive any termination of it right of access to the materials identified above.

A

B

C

D

E

F

G

H

I

L

M

N

O

P

R

S

T

U

V

W