CRSP

DRAFT

CRSP Data Definitions and Coding Schemes Guide

For the

CRSP NYSE, AMEX, Nasdaq Daily and Monthly Price and Total Return Databases

and

CRSP US Stock, Treasury Indices and Portfolio Assignments Database

Updated Annually
1925-1998

Center for Research in Security Prices
The University of Chicago Graduate School of Business

CRSP.com

CRSP Data Definitions and Coding Schemes Guide

For the

CRSP NYSE, AMEX, Nasdaq Daily and Monthly Price and Total Return Databases

and

CRSP US Stock, Treasury Indices and Portfolio Assignments Database

Covering over 22,000 Stocks, Updated Annually
1925-1998

Center for Research in Security Prices
The University of Chicago Graduate School of Business

CRSP™
The University of Chicago, Graduate School of Business
725 S. Wells Street
Suite 800
Chicago, IL 60607

Phone: 773.702.7467
Fax: 773.702.3036

e-mail: mail@crsp.uchicago.edu
http://www.crsp.com

Version CA97DD_98.01

CRSP DATA LICENSE

This is a legal agreement between you (either an individual or an entity) hereby referred to as the "user" and the University of Chicago, Graduate School of Business on behalf of the Center for Research in Security Prices, hereby referred to as "CRSP". By opening this product, you are agreeing to be bound by the terms of the following License Agreement. If you do not wish to accept these terms, return the unused, unopened data to CRSP within 30 days of receipt with any written materials to the Products and Services Department, CRSP, University of Chicago, GSB, 725 S. Wells Street, Suite 800, Chicago, IL 60607. *Opening this data without a signed agreement binds the user to restrictions of use in CRSP's standard subscription/contract terms for the product.*

The accompanying media contains data that are the property of CRSP and its information providers and are licensed for use only, by you as the original licensee. Title to such media, data and documentation is expressly retained by CRSP.

Data and documentation are provided with restricted rights. CRSP grants the user the right to access the CRSP data and the CRSP Product Database Guide(s) only for internal or academic research use. Usage of the data must be in accordance with the terms detailed in the Subscription Agreement, Contract or Agreement between CRSP and the user. The data is "in use" on a computer when it is loaded into the temporary memory (i.e. RAM) or is installed into the permanent memory (e.g. hard disk, CD ROM or any other storage device) of that computer or network in one location. **Once the data has been updated, Subscribers must promptly return the previous release of data on its original medium to CRSP or destroy it.**

COPYRIGHT NOTICE

PROPRIETARY RIGHTS

`PERMNO`, `PERMCO` and `CRSPID` are symbols representing data, which is proprietary to the Center for Research in Security Prices.

DISCLAIMER

CRSP will endeavor to obtain information appearing on its Data Files from sources it considers reliable, but disclaims any and all liability for the truth, accuracy or completeness of the information conveyed. THE UNIVERSITY AND CRSP MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH RESPECT TO THE MERCHANTABILITY, FITNESS, CONDITION, USE OR APPROPRIATENESS FOR SUBSCRIBER'S PURPOSES OF THE DATA FILES AND DATA FURNISHED TO THE SUBSCRIBER UNDER THIS SUBSCRIPTION, OR ANY OTHER MATTER AND ALL SUCH DATA FILES WILL BE SUPPLIED ON AN "AS IS" BASIS. CRSP will endeavor to meet the projected dates for updates, but makes no guarantee thereof, and shall not have any liability for delays, breakdowns or interruption of the subscription. In no event shall the University of Chicago be liable for any consequential damages (even if they have been advised of the possibility of such damages), for damages arising out of third party suppliers terminating agreements to supply information to CRSP, or for other causes beyond its reasonable control.

In the event that the Subscriber discovers an error in the Data Files, Subscriber's sole remedy shall be to notify CRSP and CRSP will use its best efforts to correct same and deliver corrections with the next update.

PERMISSION TO USE CRSP DATA

CRSP permits the use of its data in scholarly papers. CRSP data may also be used in client newsletters, marketing, and education materials, company reports, books, and other published materials. If you intend to use our data or anything derived from our data, like graphs, in your publications, we require that you receive written permission from CRSP. Please call us for a Data Permission Request at 773.702.7467.

CITATION

The following citation must be used when displaying the results of any analysis that uses CRSP data.

Source: CRSP, Center for Research in Security Prices. Graduate School of Business, The University of Chicago [year]. Used with permission. All rights reserved. crsp.com

USE OF CRSP DATABASES

PROHIBITED

Employees of academic and commercial subscribers to CRSP databases have from time to time requested the use of CRSP data for the individual's outside consulting project. CRSP databases are to be used exclusively for the subscriber's own internal educational or business processes, as stated in the Statement of Use in the subscription agreement. The University does not permit any third party to use its databases. If you wish to use our data in your consulting projects, your client may either subscribe to a CRSP database or request a custom research project.

APPROVED

The CRSP data files are proprietary and should be used only for research purposes by the faculty, students, or employees of the subscribing institution. The Subscription Agreement, signed by each subscribing institution states:

Subscriber acknowledges that the data files to which it is subscribing contain factual material selected, arranged and processed by CRSP and others through research applications and methods involving much time, study, and expense.

The Subscriber agrees that it will not transfer, sell, publish, or release in any way any of the data files or the data contained therein to any individual or third party who is not an employee or student of the Subscriber, and that the data provided to the Subscriber by CRSP is solely for the Subscriber's use.

The Subscriber may not copy the data or documentation in any form onto any device or medium without the express written consent of CRSP, except solely to create back-up copies of the CRSP data files for its internal use, subject to the terms of this Agreement.

Subscriber agrees and warrants that it will take all necessary and appropriate steps to protect CRSP's proprietary rights and copyright in the data supplied (including, but not limited to, any and all specific steps which may be expressly required by CRSP), and that the Subscriber will protect the data in no less than the manner in which it would protect its own confidential or proprietary information.

The Subscriber will inform all users and potential users of CRSP data of CRSP's proprietary rights in its files and data by giving each user a copy of this paragraph and any other specific requirements CRSP may mandate under this paragraph and by requiring each such user to comply with this paragraph and all such additional requirements.

The Subscriber agrees that its obligation under the Subscription Agreement shall survive the termination of this Agreement for any reason.

DATABASE GUIDES

Additional copies of this database guide are available on the CRSP CD-ROM and on-line through the Database Guide link found at http://www.crsp.com

OVERVIEW

ABOUT THIS GUIDE

This guide will help you to understand CRSP Stock and Indices variable definitions, coding schemes, and data organization.

Inside

Chapter One: Introduction provides an overview of CRSP and a summary of stock and indices research changes.

Chapter Two: Database Structure contains diagrams and descriptions of data structures used in the CRSP stock and indices data files.

Chapter Three: Index Methodologies describes the CRSP indices and the methodologies used to calculate them.

Chapter Four: Variable Definitions contains detailed variable definitions in alphabetical order for the CRSP stock and indices with references to utility and programming usage.

Chapter Five: CRSP Calculations contains calculations used to derive CRSP data in alphabetical order.

Chapter Six: CRSP Coding Schemes contains common coding and classification schemes used in CRSP stock data.

Appendix A: CUSIP Copyright

Appendix B: Glossary contains commonly used CRSP terminology in alphabetical order.

Index: provides an alphabetical reference to locate definitions.

Other documentation guides you may want to refer to are listed below.

See the CRSPAccess97 Installation Guide for instructions on installing data and programs on the Getting Started CD-ROM. SFA Tape Installation information can be found in Chapter 6, Data Layout of this guide.

See the CRSPAccess97 Database Format - Utilities Guide for a description of utilities available to retrieve data without programming.

See the CRSPAccess97 Database Format - Programmers Guide for information on using random access libraries and sample programs.

See the CRSP SFA Database Format Guide for information on using the SFA database format. (The database format delivered prior to 1996.)

Contact us for Technical Support
Telephone: 773.834.1025
e-mail: mail@crsp.uchicago.edu

CHAPTER 1: INTRODUCTION

OVERVIEW

This chapter provides an overview of CRSP, and research changes to the CRSP NYSE, AMEX, Nasdaq Daily and Monthly Price and Total Return Databases and the CRSP US Stock, Treasury Indices and Portfolio Assignments Database.

INSIDE

CHAPTER 1: INTRODUCTION

1.1 About CRSP

Back in 1959, Professor James Lorie fielded a call from Louis Engel, a Vice President at Merrill Lynch, Pierce, Fenner & Smith. The firm wanted to advertise how well people had done investing in common stocks, but Engel needed some solid data. Could the University of Chicago Graduate School of Business help?

That was the start of the Center for Research in Security Prices. Forty years ago, computer technology was in its infancy and no machine-readable data existed.

Professor Lorie and Professor Lawrence Fisher, a colleague on the finance faculty, set out to build a database of historical and current securities data that answered Merrill Lynch's question and, since then, many, many others.

The professors compiled the first machine-readable file. It contained month-end prices and total returns on all stocks listed on the New York Stock Exchange between 1926 and 1960. Over time, CRSP added the American Stock Exchange, the NASDAQ stock exchange, and end-of-day as well as month-end prices. Now CRSP updates US stock data in two frequencies; either once a year or once a month.

In 1999, CRSP is justly considered the best provider by far of US corporate actions information. Specifically, we diligently track name changes and name identifiers, and distributions of shares, cash, rights, spin-offs, mergers and liquidation payments. As a result, the history and quality of CRSP capital return, income return and total return numbers are unsurpassed.

CRSP Working Papers

From its founding, the University of Chicago set the highest standards of research excellence. The Graduate School of Business helped to spawn the modern revolution in finance, and research done here has been incorporated into CRSP Data Files. Among them:

Risk/Return Analysis by Harry Moskovitz
The Sharpe-Lintner Capital Asset Pricing Model
The Efficient Market Hypothesis
Black-Scholes Option Pricing Model
Small Stock Effect

The comprehensiveness and quality of CRSP data has made it the premier source for academic researchers and quantitative analysts for thirty-five years. We have the latest research on a wide variety of finance topics available online.

World Wide Web: crsp.com, CRSP Working Papers

CRSP Board of Directors

We are fortunate to have the guidance of world-renowned faculty.

Chairman Eugene F. Fama, *Robert R. McCormick Distinguished Service Professor of Finance.*
Douglas W. Diamond, *Theodore O. Yntema Professor of Finance.*
Steven Neil Kaplan, *Leon Carroll Marshall Professor of Finance*.
Robert W. Vishny, *Eric J. Gleacher Distinguished Service Professor of Finance.*
John Huizinga, *Walter David "Bud" Fackler Professor of Economics, Deputy Dean for the Faculty.*
Mark E. Zmijewski, *Professor of Accounting, Deputy Dean for the Full-Time M.B.A. Programs*.

CRSP Historical Data Products

CRSP NYSE, AMEX, Nasdaq Daily and Monthly Price and Total Return Databases

CRSP provides monthly or annual updates of end-of-day and month-end prices on all listed NYSE, AMEX and NASDAQ common stocks with basic market indices. CRSP provides the most comprehensive distribution information available, for the most accurate total return calculations.

Important facts regarding CRSP US Stock Data.

- **Annual Update**: Ready in April.
- **Monthly Updates**: Ready by the 15th day of the following month.
- **Daily and Month-End Data:** NYSE/AMEX**:** High, low, bid, ask, closing price, trading volume, shares outstanding, capital appreciation, income appreciation, total return, year-end capitalization, and year-end capitalization portfolio. NASDAQ data also includes: closing bid, ask, number of trades, historical traits information, market maker count, trading status, and NASD classification**.**
- **History:** NYSE daily data begins July 1962. Monthly data begins December 1925. AMEX daily and monthly data begins July 1962. NASDAQ daily and monthly data begins December, 14 1972.
- **Identifying Information:** Complete Name History for each security; all historical: CUSIPs, exchange codes, ticker symbols, SIC codes, share classes, share codes, and security delisting information. These items may change over time. CRSP has developed a unique permanent issue identification number, `PERMNO`, and a unique permanent company identification number, `PERMCO`. These enable the user to track the issue over time, performing extremely accurate time-series data analysis.
- **Distribution Information:** descriptions of all distributions, dividend amounts, factors to adjust price and shares, declarations, ex-distributions, record and payment dates, and security and company linking information.

CRSP US Stock, Treasury Indices and Portfolio Assignments Database

A companion database, the CRSP US Stock, Treasury Indices and Portfolio Assignments Database**,** provides market indices on a daily, monthly, quarterly and annual frequency. This database provides additional market and security level portfolio statistics and decile portfolio assignment data. Four types of indices provide the following information.

- The **CRSP Stock File Indices** includes Value- and Equal-Weighted Indices, with or without dividends, the S&P 500 Composite Index and returns, NASDAQ Composite Index and return and security data needed to link stocks to the CRSP US Market Cap-Based Portfolios. The indices files also contain the US Government Consumer Price Index, US Government Bond Fixed Term Index Series, and the CRSP Risk-Free Rates File.

- Track micro-, small-, mid- and large-cap stocks with CRSP US Market Cap-Based Portfolios. CRSP ranks all NYSE companies by market capitalization and divides them into 10 equally populated portfolios. AMEX and NASDAQ National Market stocks are then placed into deciles according to their respective capitalizations, determined by the NYSE breakpoints. CRSP Portfolios 1-2 represent large caps, Portfolios 3, 4, 5 represent mid-caps, Portfolios 6, 7, 8 represent small caps, and Portfolios 9-10 benchmark micro-caps.

  Among the monthly data provided are the number of companies in the portfolio at the start of the quarter, portfolio weight at the start of the quarter, total return and index level, capital apprecation return and index level, and income return and index level.

- **CRSP Indices for the S&P 500 Universe** are daily and monthly files which include value- and equal-weighted returns, with and without dividends.

- **CRSP US Treasury and Inflation Series** are monthly files containing returns and index levels on US Treasuries and the US Government Consumer Price Index and index level.

CRSP US Government Bills, Notes and Bonds End-of-Day and Month-End Databases

CRSP provides 1.5 million end-of-day price observations for 3,244 US Treasury bills, notes and bonds since 1961. The monthly database contains 101,986 prices for 5,136 issues since 1925. Monthly supplemental files, developed by Professor Eugene F. Fama, Professor of Finance, are described below. They are updated annually.

Important facts regarding CRSP US Treasury data.

- **Annual Updates:** Ready in April.

- **Daily Data**: Daily quote dates, delivery dates, 1-, 3-, and 6-month CD rates, 30-, 60-, and 90-day commercial paper rates, and Federal funds effective rate. **Monthly Data**: Monthly quote dates and delivery dates. Julian, linear, and other date information to facilitate date arithmetic.

- **History:** Daily data begins on June 14, 1961. Monthly data begins on December, 31 1925.

- **Identifying Information:** CRSP Identifier (`CRSPID`), CUSIP, maturity date, coupon rate, among other items, sorted by `CRSPID`.

- **Quote Data:** Bid, ask, and source. **Performance Data**: Accrued interest, yield, return and duration.

- **Debt Data**: Debt outstanding, total and publicly held.

- **Fixed Term Indices Files:** Performance of single Treasury issues at fixed maturity horizons.

- **Supplemental Files:** The CRSP **US Treasury Securities Month-End Database** contains files designed by Eugene F. Fama, Professor of Finance, The University of Chicago Graduate School of Business. These files extract term structures and risk-free rates. There are four groups of files. The Treasury Bill Term Structure Files, The Fama-Bliss Discount Bond Files, The Risk-Free Rates File and The Maturity Portfolio Returns File. The data in these files begin in 1952 with the exception of the Risk-Free Rates File, where the data begins in 1925.

CRSP Survivor-Bias Free US Mutual Fund Database

based on the Standard & Poor's® Micropal® Database

In estimating the performance on an equal-weighted index of equity mutual funds, Mr. Carhart found that, "Using only surviving funds biases these (performance) measures upward by about one percent per year."

Recently introduced, the **CRSP Survivor-Bias Free US Mutual Fund Database** records each mutual fund's name and organizational history. CRSP tracks monthly returns, monthly total net assets, monthly net asset values and monthly distributions for open-ended mutual funds from January 1, 1962, to December 31, 1997. Updated quarterly, the database uses Microsoft Access 97 database software.

Mark M. Carhart developed this unique database for his 1995 dissertation submitted to the Graduate School of Business entitled, *Survivor Bias and Persistence in Mutual Fund Performance.* In it he noted that the explosion in new mutual funds has been "accompanied by a steady disappearance of many other funds through merger, liquidation and other means. …this data is not reported by mutual fund data services or financial periodicals and in most cases is (electronically) purged from current databases. This imposes a selection bias on the mutual fund data available to researchers: only survivors are included."

Sample Data Sets

Sample data sets for all CRSP products are available on the Getting Started CD-ROM.

1.2 Changes to the 1998 CRSPAccess97 Stock Files

CRSP has discontinued the following products:

Product Code	Name
DX	Daily NYSE/AMEX History
MZ	Monthly NYSE/AMEX History
PF	Pricing bundle of DX & MZ

Variable Name Changes

The variables, previously organized by programming array, have been reorganized to facilitate user access. The variables are listed alphabetically by name, and contain a table following their description containing a direct maps of their usage for programming in FORTRAN and C, and for the CRSPAccess97 *ts_print* and *stk_print* utility programs.

The CRSPAccess97 variable typename in the C indices documentation has been changed to groupname.

Delisting Research

Delisting codes have been updated, missing distributions and delisting returns have been added, and some existing delisting returns have been adjusted.

Dropped Issues: Research has been completed for approximately 800 delistings that were coded as "Dropped" in the CRSP Stock Files as follows.

- The issue dropped from the New York Stock Exchange or the American Stock Exchange after 1962.
- The delisting code was between 500 and 588.
- The issue had a missing delisting return (`dlret` = -55, -66, or -99.)

New delisting codes for "dropped" issues and announcement information have been have been added. See Chapter 6 for additional information.

CRSP guidelines were established for accepting prices and distributions for delisting returns calculations. (See Delisting Structure and Delisting Return in the Concepts Section.)

Of approximately 800 delistings that were researched, 127 new delisting returns were calculated for the daily file and an additional 120 for the monthly file. 247 new returns dating from 1962 on NYSE and AMEX previously missing have been updated in this data release.

Distribution Research

New Delisting Codes: A new distribution code 7*** has been added to include dropped issue announcements.

The following codes have been added to the data file: 3884, 5538, 5872, 7111, 7121, 7131, 7141, 7151, 7161, 7171, and 7181.

Periodic Distributions: Data between 1995 – 1998 for missing periodic cash distributions (annual, semi-annual, quarterly, and monthly dividend payments.) has been researched. A few missing distributions, were added and a few extraneous distributions, were deleted. Mis-codings were corrected.

Discontinued Codes: 6*** codes, excepting tender issue (6225) and right's issuance (6541), are no longer assigned as of May 1997 data.

Program Enhancements

***ts_print*:** Program enhancements and new features and options have been added to *ts_print* utility. See the CRSPAccess97 Database Format - Utilities Guide for additional details.

Significant Data Edits

The following corrections have been made.

PERMNO 45356 (Tyco Labs, Inc.) merged with PERMNO 76805 (ADT, Ltd.) in July of 1997, forming Tyco International, Ltd. (PERMNO 45356). It was incorrectly listed under PERMNO 76805. This was corrected in the October 1998 release of the data.

PERMCO 11369 (F & M Bancorp, MD.) was incorrectly assigned to PERMNO 79635 (F & M Bancorp, Inc.). PERMCO 34874 has been assigned to F&M Bancorp, Inc. It has been corrected in the October 1998 release of the data.

PERMNO 86217 (Valero Natural Gas Partners LP) was previously omitted from the subscriber database. It was available in the July 1998 release of the data.

PERMNO 76789 (CFB Capital I Securities) had the wrong name in the file. It has been corrected to Community First Bankshares, Inc. in the July 1998 release of the data.

PERMNO 34527 (Federated Investors, Inc) has been edited and was available in the June 1998 release of the data as PERMNO 86102 (Federated Investors, Inc, PA).

PERMNO 75819 (Vencor, Inc.) split. The history from May, 1998 is available under PERMNO 86103 (Vencor, Inc. New). Data prior to May, 1998 was merged into the history for PERMNO 86040 (Ventas, Inc.)

PERMNO 85657 (Promus Hotel Corp New) merged with PERMNO 81671 (Promus Hotel Corp New) in the February 1998 release of the data. The history of 85657 is now available under 81617.

PERMNO 85920 (Ocean Energy, Inc) merged with PERMNO 81153 (Ocean Energy, Inc) in the April 1998 release of the data. The history of 85920 is now available under 81153.

PERMNO 85681 (C F B Capital II), a preferred, began trading in December of 1997 and was assigned an incorrect share code. This was corrected in the April 1998 release of the data.

PERMNO 84663 (Hertz Corp) was incorrectly assigned PERMCO 21856 (Valspar Corp). The correct PERMCO 34736 was assigned in the May 1998 data release.

PERMNO 82991 (Williams Industries, Inc) was missing 10 years of data in the March 1998 and April 1998 data releases. This was corrected on the May 1998 data release.

PERMNO 75967 (Amcor Capital Corp) 1998 prices were incorrectly attached to PERMNO 12598 (ACAP Corp) on the data releases between January 1998 and April 1998. This has been corrected on the May 1998 data release.

The following issues have been added to the data file:

PERMNO	Name
78640	Celcor, Inc.
78615	Columbian Energy Co. Ltd Partners
78625	Family Group Broadcasting LP
78849	Talcon LP
77013	United Realty Group LP
77254	American Ins Mtg Invs LP Ser 85
75059	Fine Homes International LP
75180	Pimco Advisors Holdings LP

75321	Universal Medical Buildings LP
12000	National Income Realty Trust
78387	Daleco Resources, Inc.
85619	Meditrust Corp.
86104	Burlington Resources, Inc. CSGR
86105	Enterprise Oil PLC
86106	Lakehead Pipeline Partners LP
86210	Broughton Foods, Co.
86214	Sky Network Television, Ltd.
86441	International Manu Srvcs, Inc.

The following issues have been deleted from the file:

75210
77996
79317
81664
81768
81915
82630
84336
85070
85628
85657

CHAPTER 2: CRSP DATABASE STRUCTURE

OVERVIEW

This chapter contains the layout and organization of the CRSP stock and indices data files.

INSIDE

CHAPTER 2: CRSP DATABASE STRUCTURE

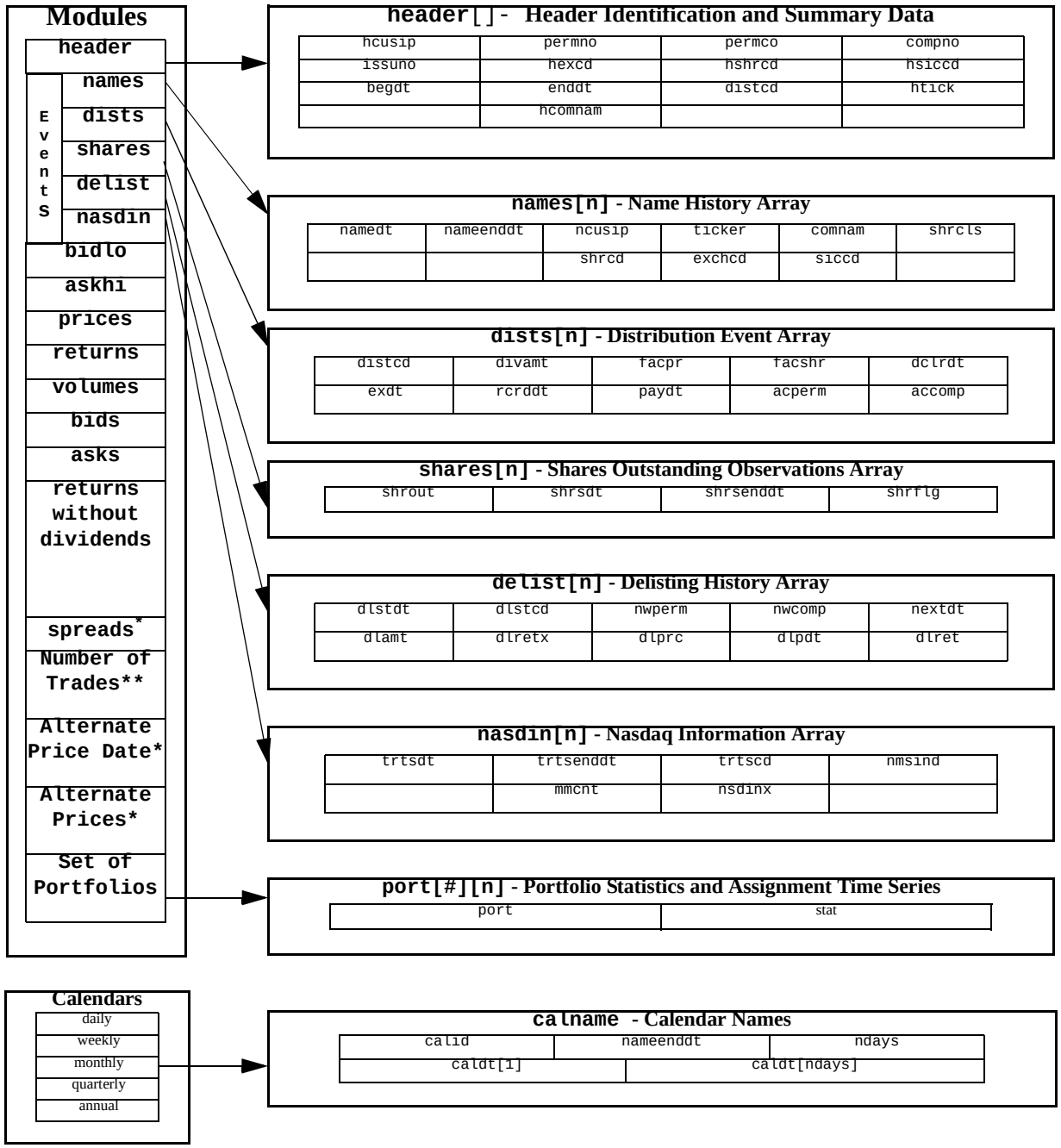
This section contains general and background information about the CRSP data and variables and diagrams to depict the structures. The CRSP data is organized in two different database formats, SFA and CRSPAccess97. The SFA database is designed for sequential FORTRAN access of the data. CRSPAccess97 is designed for random or sequential C or FORTRAN access, as well as non-programming or utility program access.

See the CRSPAccess97 Database Format - Programmers Guide for details on programming use.

2.1 CRSP Database Structure Diagrams

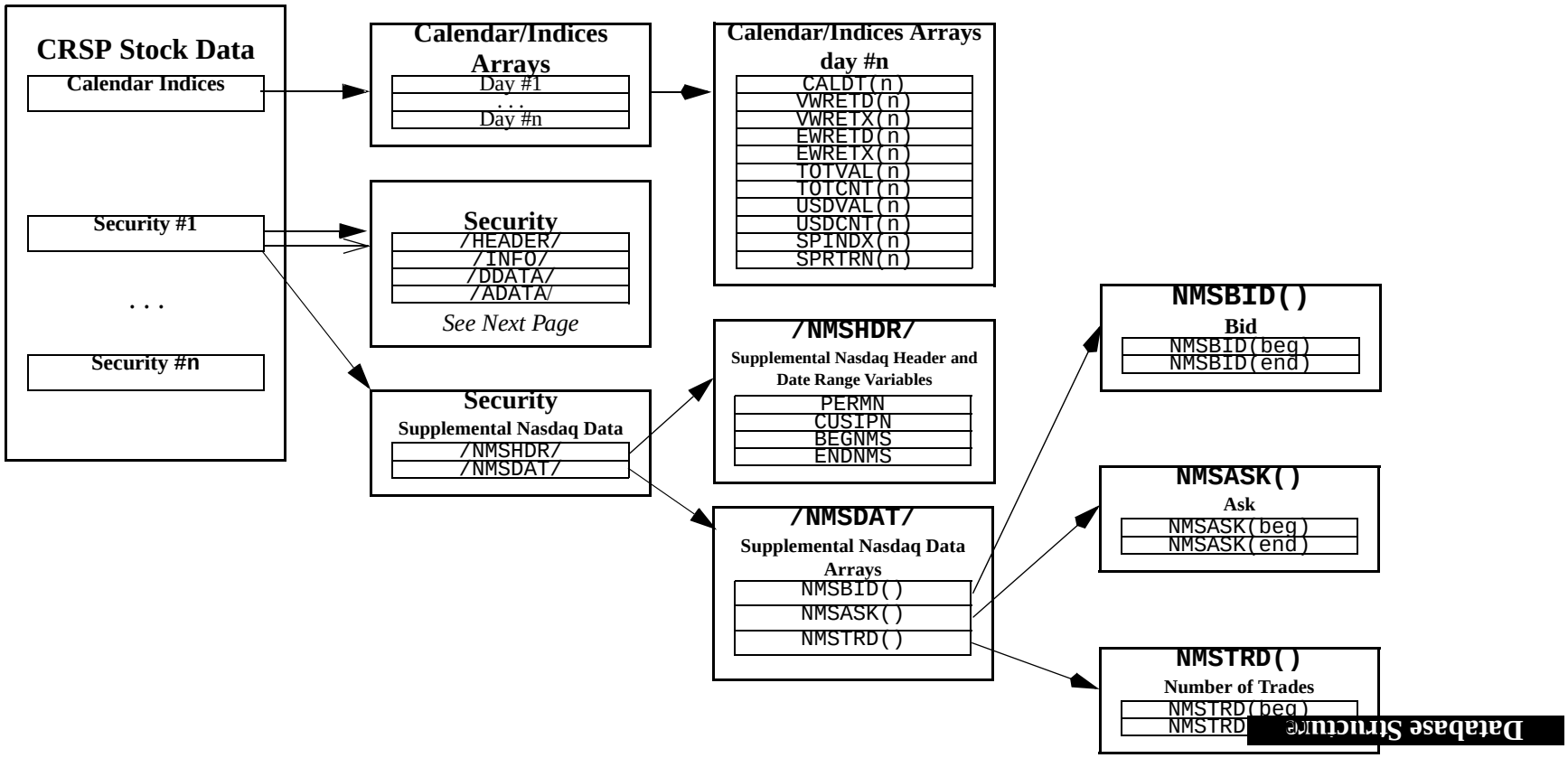
The database structure of the stock and indices files in CRSPAccess97 and SFA formats are organized by format and programming use. The variables are listed by mnemonic in each diagram. There is not sufficient room in the diagrams to list the variable's name, so each section in 2.4 includes the portion of the diagram that pertains to the description and has the name with the mnemonic. The variables are defined in the Data Definitions section of this guide. (See "Chapter 4: Data Definitions" on page 55).

CRSP Stock Data Layout - CRSPAccess97 - C Usage

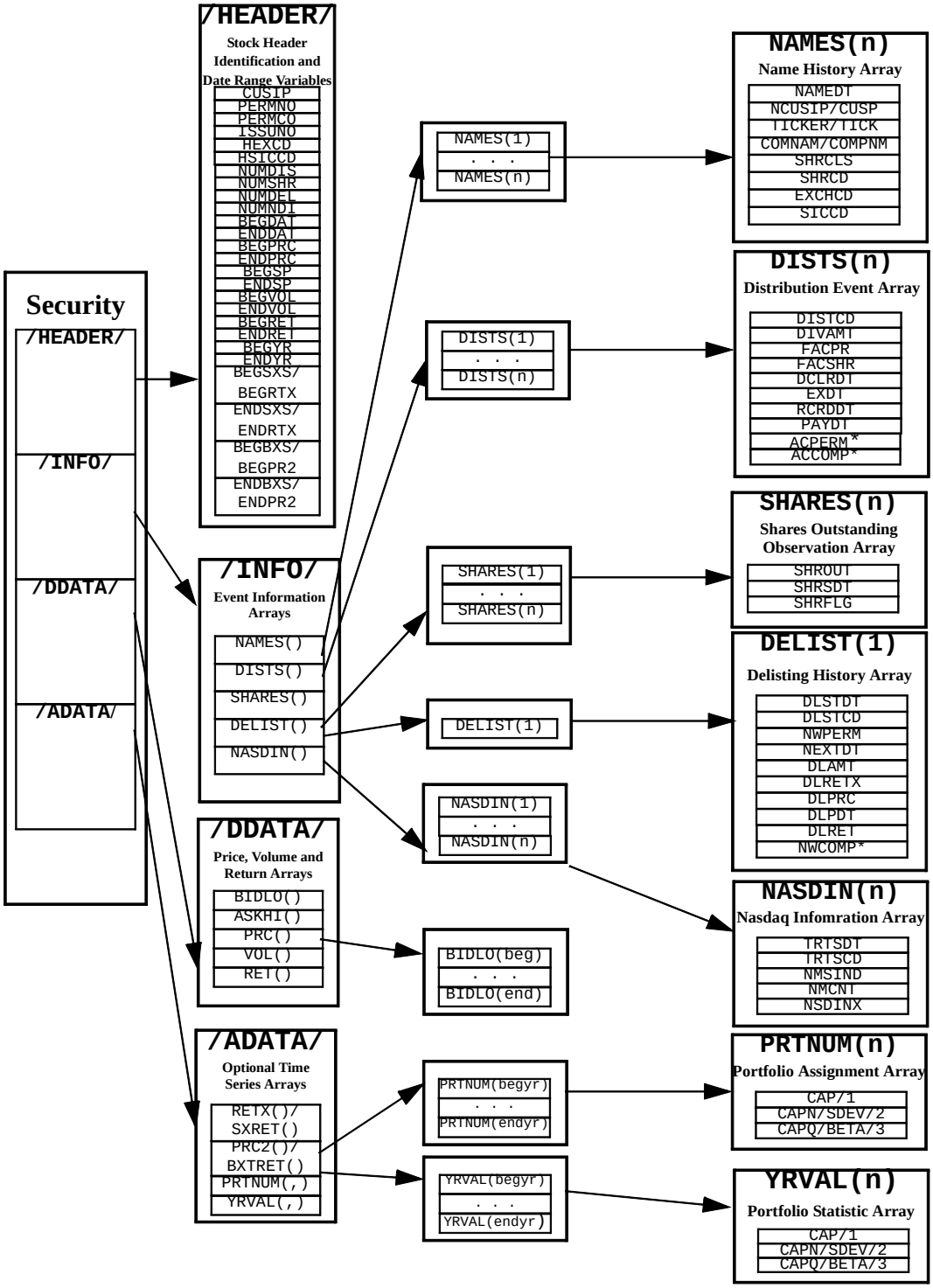


* - Monthly Data Only
** - Daily Data Only

CRSP Stock Database Layout - SFA, CRSPAccess97 - FORTRAN 77 Usage

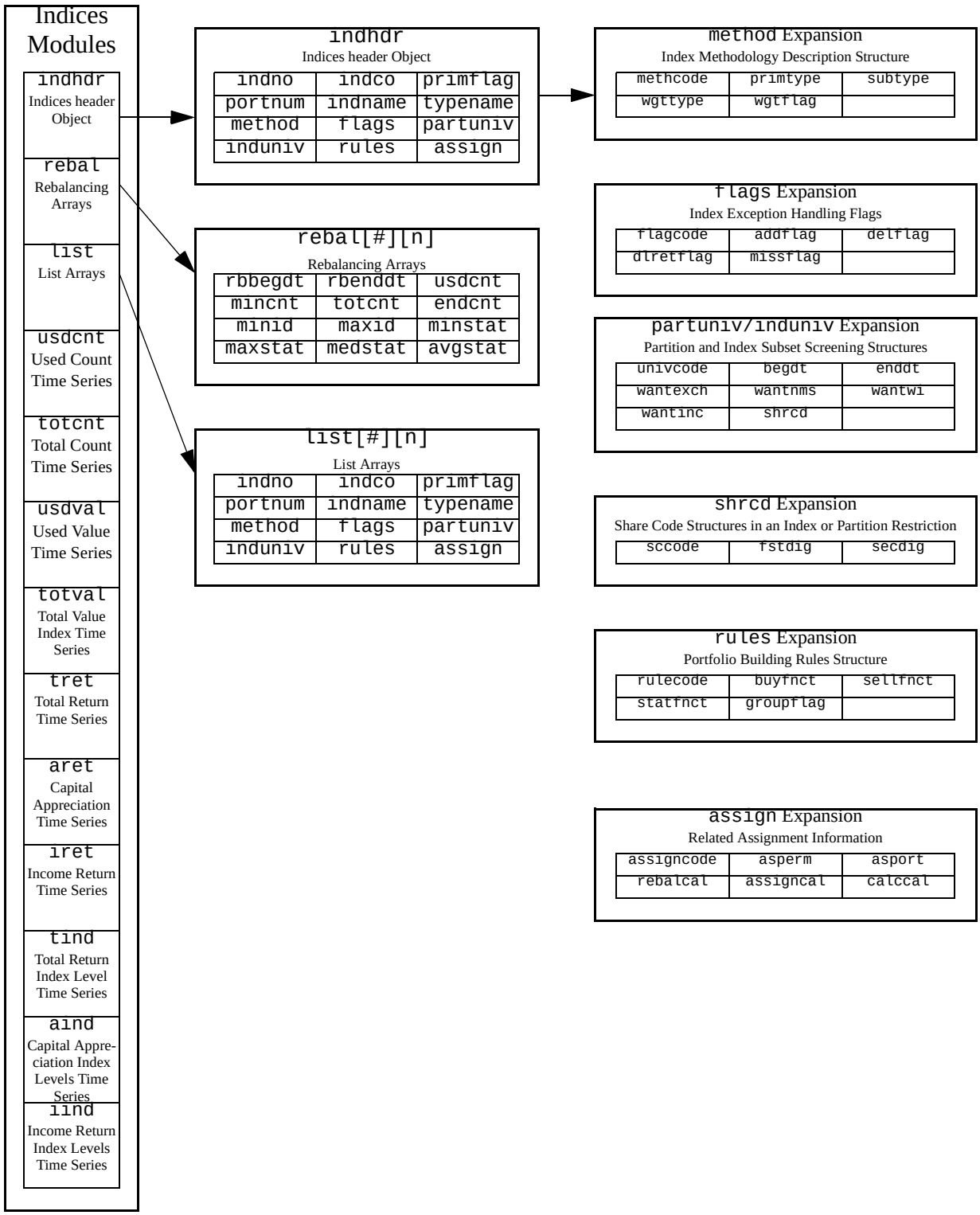


Stock Database Layout - FORTRAN 77 (Con't)



*CRSPAccess97 Only

CRSP Indices Data Layout - CRSPAccess97 - C Usage



CRSP Indices Data Layout - CRSPAccess97, SFA - FORTRAN 77 Usage

CRSP Stock File Indices Array

CALDT	VWRETD	VWINDD	VWRETX
VWINDX	EWRETD	EWINDD	EWRETX
EWINDX	NCINDX	NCRTRN	DECRET
DECIND	TOTVAL	TOTCNT	

CRSP Cap-Based Monthly History Array

CALDT	PRTNAM	PRTCNT	PRTWGT
TOTRET	TOTIND	CAPRET	CAPIND
	INCRET	INCIND	

CRSP Cap-Based Reports Rebalancing History Array

YYYYMM	PRTNO	PRTCCT	MINCWT
MINCNM	MAXCWT	MAXCNM	

CRSP CTI Indices Array

CALDT	B30RET	B30IND	B20RET
B20IND	B10RET	B10IND	B7RET
B7IND	B5RET	B5IND	B2RET
B2IND	B1RET	B1IND	T90RET
T90IND	T30RET	T30IND	CPIRET
	CPIIND		

2.2 Stock and Indices Data Strucures

Stock Data Structures

This section describes each stock data array included in the database, organized for C or FORTRAN access. Accessibility and data structures are different in some arrays between C and FORTRAN. When they differ, a description and layout is included for each. Please refer to the diagrams in sections 2.1 - 2.3, 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 CRSPAccess97 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 is 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. See Header Identification and Date Range Variables for FORTRAN Stock Header details.

Header Identification and Summary Data fields include:

- Primary Permanent Identifiers, PERMNO, and PERMCO,
- Secondary Permanent Identifiers, CUSIP Identifier, Header, Nasdaq Company Number, Nasdaq Issue Number,
- Most recent Name History Array information Exchange Code, Header, Share Code, Header, Standard Industrial Classification (SIC) Code, Header, Ticker Symbol, Header, and Company Name, Header,
- Most recent listing information, Delisting Code, Header, and
- Security data ranges in the file, Begin of Stock Data, and End of Stock Data.

`header[]` Header Identification and Summary Data

CUSIP, Header (`hcusip`)	PERMNO (`permno`)	PERMCO (`permco`)	Nasdaq Company Number (`compno`)
Nasdaq Issue Number (`issuno`)	Exchange Code, Header (`hexcd`)	Share Code, Header (`hshrcd`)	Standard Industrial Classification (SIC) Code, Header (`hsiccd`)
Begin of Stock Data (`begdt`)	End of Stock Data (`enddt`)	Distribution Code, Header (`distcd`)	Ticker Symbol, Header (`htick`)
	Company Name, Header (`hcomnam`)		

Header Identification and Date Range Variables

Header Identification and Date Range Variables are a set of variables in a CRSP stock database or accessed using CRSP FORTRAN access programs in both the CRSPAccess97 and SFA database formats. In this format, the ranges of valid data for a security for all event arrays and all time series are included in the central header record. Primary and secondary identifiers and selected most recent exchange and industry classifications are included.

The Header Identification and Date Range Variable fields include:

- Primary Permanent Identifiers PERMNO, and PERMCO,
- Secondary Permanent Identifiers, CUSIP, Identifier Header, and Nasdaq Issue Number,
- Most recent name information, Exchange Code, Header, Standard Industrial Classification (SIC) Code, Header,
- Counts of array events, Number of Name Structures, Number of Distribution Structures, Number of Delisting Structures, Number of Shares Structures, Number of Nasdaq Information Structures,
- Ranges of time series data, Begin Index of Stock Data, End Index of Stock Data, Begin Index of Price Data , End Index of Price Data, Begin Index of Return Data, End Index of Return Data, Begin Index of Spread between Bid and Ask, End Index of Spread between Bid and Ask, Begin Index of Return without Dividends Data, End Index of Return without Dividends Data, Begin Index of Beta Excess Return Data, End Index of Beta Excess Return Data, Begin Index of Optional Time Series 2 Data, End Index of Optional Time Series 2 Data, Begin Index of Nasdaq Data, End Index of Nasdaq Data, Begin Index of Portfolio Data, End Index of Portfolio Data, Begin Index of Volume Data, End Index of Volume Data, Begin Index of Secondary Price Data, End Index of Secondary Price Data, Begin Index of Standard Deviation Excess Return Data, End Index of Standard Deviation Excess Return Data, Begin Index of Optional Time Series 1 Data, and End Index of Optional Time Series 1 Data.

/HEADER/ Header Identification and Date Range Variables

CUSIP, Header (CUSIP)	PERMNO (PERMNO)	PERMCO (PERMCO)	Nasdaq Issue Number (ISSUNO)	Exchange Code, Header (HEXCD)
Standard Industrial Classification Code, Header (HSICCD)	Number of Name Structures (NUMNAM)	Number of Distribution Structures (NUMDIS)	Number of Shares Structures (NUMSHR)	Number of Delisting Structures (NUMDEL)
Number of Nasdaq Information Structures (NUMNDI)	Begin Index of Stock Data (BEGDAT)	End Index of Stock Data (ENDDAT)	Begin Index of Price Data (BEGPRC)	End Index of Price Data (ENDPRC)
Begin Index of Secondary Price Data (BEGSP)	End Index of Secondary Price Data (ENDSP)	Begin Index of Volume Data (BEGVOL)	End Index of Volume Data (ENDVOL)	Begin Index of Return Data BEGRET
End Index of Return Data (ENDRET)	Begin Index of Portfolio Data (BEGYR)	End Index of Portfolio Data (ENDYR)	Begin Index of Standard Deviation Excess Return Data* (BEGSXS)	End Index of Standard Deviation Excess Return Data* (ENDSXS)
Begin Index of Optional Time Series 1 Data* (BEGSXS)	End Index of Optional Time Series 1 Data* (ENDSXS)	Begin Index of Return without Dividends Data* (BEGRTX)	End Index of Returns without Dividends Data* (ENDRTX)	Begin Index of Beta Excess Return Data‡ (BEGBXS)
End Index of Beta Excess Return Data‡ (ENDBXS)	Begin Index of Optional Time Series 2 Data‡ (BEGBXS)	End Index of Optional Time Series 2 Data‡ (ENDBXS)	Begin Index of Spread between Bid and Ask Data‡ (BEGPR2)	End Index of Spread between Bid and Ask Data‡ (ENDPR2)

*3 variable options are listed for the same data slot, of which only one may be loaded.
‡3 variable options are listed for the same data slot, of which only one may be loaded.

Supplemental Nasdaq Header and Date Range Variables

The Supplemental Nasdaq Header contains header information for the Supplemental Nasdaq Time Series on Supplemental Nasdaq files in SFA format.

- Redundant identifiers available on SFA files used to match the Supplemental Nasdaq File to the Stock File (PERMNO, Nasdaq and CUSIP, Nasdaq).
- Nasdaq Time Series range information (Begin Index of Nasdaq Data, End Index of Nasdaq Data).

/NMSHDR/ Supplemental Nadsaq Header and Date Range Variables

PERMNO, Nasdaq (PERMN)	CUSIP, Nasdaq (CUSIPN)	Begin Index of Nasdaq Data (BEGNMS)	End Index of Nasdaq Data (ENDNMS)

Calendar/Index Arrays

The Calendar/Index Arrays are a set of arrays containing calendar and index data in the SFA format. In this format, a calendar is provided that can be used with all time series data in a stock file except portfolio data. Selected data for three different indices, the CRSP value- and equal-weighted market indices, and a S&P 500 composite index are included.

The Calendar/Indices Arrays furnish eleven data arrays, containing:

- Calendar Trading Dates,
- Returns on a Value-Weighted Index (Value-Weighted Return Index (Including all Distributions) and Value-Weighted Return Index (Excluding Dividends)),
- Returns on an Equal-Weighted Index (Equal-Weighted Return Index (Including all Distributions) and Equal-Weighted Return Index (Excluding Dividends)),
- Market counts (Total Count of Market and Count of Securities Used)
- Market weights (Total Value of Market and Market Value of Securities Used)
- S&P 500 Composite Index Level or Index Level on Nasdaq Composite
- S&P 500 Composite Index Return or Return on Nasdaq Composite Index

Calendar/Index Array

Calendar Trading Date (CALDT)	Return (Including all Distributions) Value-Weighted Index (VWRETD)	Return (Excluding Dividends) on Value-Weighted Index (VWRETX)	Return (Including all Distributions) on Equal-Weighted Index (EWRETD)
Return (Excluding Dividends) on Equal-Weighted Index (EWRETX)	Total Value of Market (TOTVAL)	Total Count of Market (TOTCNT)	Market Value of Securities Used (USDVAL)
Count of Securities Used (USDCNT)	S&P 500 Composite Index Level (SPINDX) or Index Level on Nasdaq Composite (NCINDX)	S&P 500 Composite Index Return (SPRTRN) or Return on Nasdaq Composite Index (NCRTRN)	

On daily files, the calendar lists all NYSE/AMEX and Nasdaq trading dates from July 2, 1962 through the end of the file. The beginning date for Nasdaq data is day 2610, which corresponds to December 14, 1972. This calendar includes only active market days and thus excludes weekends and market holidays. The monthly index file has all month-end trading days on the NYSE from December 1925 through the last month of the file. The AMEX and Nasdaq monthly data begins in month 440, July 1962, and in month 565, December 1972, respectively.

In the CRSP Daily and Monthly Stock Files, the Market Indices for NYSE/AMEX/Nasdaq and the levels and returns on the Standard & Poor's 500 Composite Index are provided. Other indexes including the Nasdaq Composite can be loaded using CRSPAccess97 to SFA conversion utilities or CRSPAccess97 FORTRAN access functions if the CRSP US Stock, Treasury Indices and Portfolio Assignments Database is available.

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. The Name History Array is CRSPAccess97 and SFA database formats, but *Last Date of Name* is only available in CRSPAccess97.

Each name history observation contains the following variables:

- Names Effective Date and Last Date of Name
- Company Name
- CUSIP
- Exchange Code
- Share Code
- Ticker Symbol
- Share Class
- Standard Industrial Classification (SIC) Code

`names[]` Name History Array

Name Effective Date (`namedt`)	Last Date of Name* (`nameenddt`)	CUSIP (`ncusip`)	Ticker Symbol (`ticker`)	Company Name (`comnam`)	Share Class (`shrcls`)
		Share Code (`shrcd`)	Exchange Code (`exchcd`)	Standard Industrial Classification (SIC) Code (`siccd`)	

* CRSPAccess97 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 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 to shareholders for a security. Each distribution event includes:

- Codes describing the event (Distribution Code),
- Cash value per share (Dividend Cash Amount),
- Factor to Adjust Price,
- Factor to Adjust Shares,
- Dates associated with the distribution (Distribution Declaration Date, Ex-Distribution Date, Record Date, and Payment Date), and
- Links to securities or companies related to the event (Acquiring PERMNO*, Acquiring PERMCO.

`dists[n]` Distribution Event Array

Distribution Code (`distcd`)	Dividend Cash Amount (`divamt`)	Factor to Adjust Price (`facpr`)	Factor to Adjust Shares Outstanding (`facshr`)	Distribution Declaration Date (`dclrdt`)
Ex-Distribution Date (`exdt`)	Record Date (`rcrddt`)	Payment Date (`paydt`)	Acquiring PERMNO(`acperm`)*	Acquiring PERMCO* (`accomp`)

*CRSPAccess97 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 events 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, and related dates of all splits are provided.
- Spin-offs – All spin-offs 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 of 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 161, for the coding scheme used by CRSP. See “Distribution Events Table” on page 163 for examples of specific cases of distributions.

Shares Outstanding Observations Array

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

Each observation array contains the following fields:

- Shares Observation Date
- Shares Observation Ending Date*
- Shares Outstanding
- Shares Flag Code

`shares[]` Shares Outstanding Observation Array

Shares Outstanding (`shrout`)	Shares Observation End Date* (`shrsenddt`)	Shares Outstanding Observation Date (`shrsdt`)	Shares Outstanding Observation Flag (`shrflg`)

* CRSPAccess97 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 to calculate market capitalization.

Delisting History Array

The Delisting History 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 or Nasdaq, and delisting return. Active issues have a delisting history event where Delist Date is set to the last date of available 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.

The Delisting History Array events contain the following information:

- Delisting Date,
- Delisting Code,
- Prices found after delisting (Delisting Date of Next Available Information and Delisting Price)
- Links to securities or companies that can be used to track the issue further (New PERMNO, New PERMCO),
- Share Values after delisting (Amount after Delisting and Delisting Payment Date)
- Delisting Return and Delisting Return without Dividends

Delisting History Array

Delisting Date (dlstdt)	Delisting Code (dlstcd)	New PERMNO (nwperm)	New PERMCO* (nwcomp)	Delisting Date of Next Available Information (nextdt)
Amount After Delisting (dlamt)	Delisting Return without Dividends (dlretx)	Delisting Price (dlprc)	Delisting Payment Date (dlpdt)	Delisting Return (dlret)

* CRSPAccess97 C and FORTRAN only.

In current CRSP files only the most recent delisting event is coded in the Delisting History 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 is untracked, and time series data is filled in with missing values.

Delisting information is based on the exchange of shares at the earliest possible opportunity, by trade on a secondary market, payments from the company, or outstanding tender offer. If information is not available immediately 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 apply 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.**Group History Arrays**

Group History Arrays are a set of events array structures containing data on the membership of securities within

predefined groups.

Each Group List History array element contains the following information:

- Date ranges when group information is applicable (First Date Included in Group and Last Date Included in Group), and
- Grouping Flags (Group Flag of Associated Index and Group Subcategory Flag)

The variable Number of Group Types contains the count of group arrays available for all securities in a set. There are four possible group arrays in current stock databases, but these have yet been defined. Each array has its own count, which is set to zero if not applicable to the particular security.

No group histories are available in the current stock database.

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 has at least one Nasdaq Information Array.

The following fields are available in the Nasdaq Information Array structures:

- Nasdaq Traits Date
- Nasdaq Traits End Date
- Nasdaq Traits Code
- Nasdaq National Market Indicator
- Market Maker Count
- NASD Index Code

`nasdin[]` Nasdaq Information Array

Nasdaq Traits Date (`trtsdt`)	Nasdaq Traits End Date (`trtsenddt`)	Nasdaq Traits Code (`trtscd`)
Nasdaq National Market Indicator (`nmsind`)	Market Maker Count (`mmcnt`)	NASD Index Code (`nsdinx`)

This information is supplied by the National Association of Securities Dealers, Inc. (NASD). Nasdaq information structures are available for securities trading on Nasdaq beginning on April 1, 1982 for traits date and The 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 are subject to last-sale reporting. In June of 1992 the regular Nasdaq segment of The Nasdaq Stock Market[SM] was renamed The Nasdaq Small-Cap 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. `TRTSDT`, `TRTSCD`, and `NMSIND` are complete. All other fields are missing.

Price, Volume, and Return Arrays

Price, Volume, and Return 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). Correlative Supplemental Nasdaq data is stored in the daily SFA Supplemental Nasdaq time series array.

The Price, Volume, and Return Arrays are:

- Price or Bid/Ask Average
- Bid or Low Price
- Ask or High Price
- Volume Traded
- Holding Period Total Return and Return without Dividends

Bid or Low Price (bidlo)	Ask or High Price (askhi)	Price or Bid/Ask Average (prc)	Holding Period Total Return (ret)	Volume Traded (vol)

Auxiliary Time Series Data

Auxiliary Data Time Series are additional time series provided in CRSPAccess97 Stock Files.

The Auxiliary Data Time Series are:

- Returns without Dividends
- Spread Between Bid and Ask (only available in monthly databases)
- Price Alternate (only available in monthly databases)
- Price Alternate Date (only available in monthly databases)

Return without Dividends (retx)	Spread between Bid and Ask (spread-C) (PRC2-FORTRAN)	Price Alternate (altprc)	Alternate Price Date - monthly only (numtrd)

Optional Time Series Array Data

A stock file in SFA format or accessed with CRSP FORTRAN access can access two optional time series variables from a list including Auxiliary Data Time Series, Supplemental Nasdaq Time Series, and derived excess returns time series. There are three variable names for each of these time series slots:

- Optional Time Series 1 (Returns without Dividends or Standard Deviation Excess Returns)
- Optional Time Series 2 (Spread of Bid and Ask or Beta Excess Returns)

Standard Deviation Excess Returns (SXRET) Returns without Dividends (RETX) Optional Time Series 1 (SXRET)	Beta Excess Returns (BXRET) Spread of Bid and Ask (PRC2) Optional Time Series 2 (BXRET)

The data loaded to the SFA Optional Time Series is determined by access parameters in a CRSPAccess97 to SFA Conversion Utility (see the CRSP SFA Database Format Guide) or set in a CRSPAccess97 FORTRAN program. The default is to load Returns without Dividends to Optional Time Series 1 and Spread of Bid and Ask to Optional Time Series 2. Excess returns can be calculated and loaded to these time series based on a selected portfolio types if the CRSP Indices and Portfolio Assignments File is available.

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 and contains a history of statistics and portfolio assignments for a security. Two variables are available for each calendar period:

- Portfolio Assignment Number and
- Portfolio Statistic Value

Portfolio Statistics and Assignment Time Series	
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, and 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 of an index that contains the performance results of the 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 Indices File and Portfolio Assignment product. Indices based on the portfolios formed for these Portfolio Types are also available in the CRSP Indices File and Portfolio Assignments product.

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

Supplemental Nasdaq Time Series

Supplemental Nasdaq Time Series contain additional time series data for Nasdaq securities. Supplemental Nasdaq Time Series data is available starting in November, 1982, and all Nasdaq Small-Cap Securities since June 15, 1992. Due to data source limitations data is missing for 15 Nasdaq National Market securities in December, 1982, and for all the Nasdaq National Market securities in February, 1986.

The following time series are available:

- Bid
- Ask
- Number of Trades, Nasdaq

Bid (bid)	Ask (ask)	Number of Trades, Nasdaq (numtrd)

The Supplemental Nasdaq Time Series are integrated in the CRSPAccess97 daily and monthly stock files. They are provided on an additional tape file in SFA format with Daily Stock Files. There are no CRSPAccess97 conversion utilities available to generate this SFA format directly. However, the fields can be loaded directly into Optional Time Series fields in SFA Stock Files with CRSPAccess97 to SFA conversion utilities.

Portfolio Assignment Array

Portfolio Assignment Array is a set of time series containing security portfolio assignment data in a CRSP stock database in SFA Format or accessed using CRSP FORTRAN access programs. In this format, the assignment data for three portfolio types can be loaded. All use an annual rebalancing calendar and share the same data ranges specified by the Begin Index of Portfolio Data and the End Index of Portfolio Data.

Each of the three slots can be loaded with portfolio assignment data from the list of available portfolio types. More than one variable name exists for each slot since different data can be loaded. The primary variable name is used for the default. The variable names are:

- Portfolio Assignment for First Portfolio (Portfolio Assignment for Capitalizations)
- Portfolio Assignment for Second Portfolio (Portfolio Assignment for Standard Deviations or Portfolio Assignment for NYSE/AMEX Capitalizations)
- Portfolio Assignment for Third Portfolio (Portfolio Assignment for Betas or Portfolio Assignment for Nasdaq Capitalizations)

Portfolio Assignment Array

Portfolio Assignment for Betas (BETA)	Portfolio Assignment for Capitalizations (CAP)	Portfolio Assignment for First Portfolio (CAP)
Portfolio Assignment for Nasdaq Capitalizations (CAPQ)	Portfolio Assignment for NYSE/ AMEX Capitalizations (SDEV)	Portfolio Assignment for Second Portfolio (SDEV)
Portfolio Assignment for Standard Deviations (SDEV)	Portfolio Assignment for Third Portfolio (BETA)	

Each element in the time series arrays represents one year of data. Instead of using the calendar available in the Calendar/Indices Arrays, the applicable year is found by adding 1924 to the index of the data.

In a Monthly or Daily stock file only the NYSE/AMEX/Nasdaq Capitalization Deciles are available and loaded by default to Portfolio Assignment for First Portfolio. The other two fields contain only missing values. If the Indices and Portfolio Assignments File is also available, all annual portfolio types can be loaded. By default, NYSE/AMEX Capitalizations are loaded to Portfolio Assignment for Second Portfolio and Nasdaq Capitalizations are loaded to Portfolio Assignment for Third Portfolio. See Index Methodology for the available portfolio types.

Portfolio Statistic Array

The Portfolio Statistic Array is a two-dimensional array containing security portfolio statistic data in a CRSP stock database in SFA Format or accessed using CRSP FORTRAN access programs. In this format, the statistic data for three portfolio types can be loaded. All are based on an annual rebalancing calendar and share the same data ranges specified by the Begin Index of Portfolio Data and The End Index of Portfolio Data.

Each of the three slots can be loaded with portfolio statistics data from the list of available portfolio types. More than one variable name exists for each slot since different data can be loaded. The primary variable name is used for the default. The variable names are:

- Portfolio Statistic for First Portfolio (Portfolio Statistic for Capitalizations)
- Portfolio Statistic for Second Portfolio (Portfolio Statistic for Standard Deviations or Portfolio Statistic for NYSE/AMEX Capitalizations)
- Portfolio Assignment for Third Portfolio (Portfolio Statistic for Betas or Portfolio Statistic for Nasdaq Capitalizations)

Portfolio Statistic Array

Portfolio Statistic for Betas (BETA)	Portfolio Statistic for Capitalizations (CAP)	Portfolio Statistic for First Portfolio (CAP)
Portfolio Statistic for Nasdaq Capitalizations (CAPQ)	Portfolio Statistic for NYSE/AMEX Capitalizations (SDEV)	Portfolio Statistic for Second Portfolio (SDEV)
Portfolio Statistic for Standard Deviations (SDEV)	Portfolio Statistic for Third Portfolio (BETA)	

Each element in the time series arrays represents one year of data. Instead of using the calendar available in the Calendar/Indices Arrays, the applicable year is found by adding 1924 to the index of the data.

In a Monthly or Daily stock file only the NYSE/AMEX/Nasdaq Capitalization Deciles are available and loaded by default to Portfolio Statistic for First Portfolio. The other two fields contain only missing values. If the Indices and Portfolio Assignments File is also available, all annual portfolio types can be loaded. By default, NYSE/AMEX Capitalizations are loaded to Portfolio Statistic for Second Portfolio and Nasdaq Capitalizations are loaded to Portfolio Statistic for Third Portfolio. See Index Methodology for the available portfolio types.

CRSPAccess97 C Index Data Structures

There are four types of indices provided with the CRSPAccess97 Indices Files. In additional, security portfolio assignmetn data are porvided in asosciation with market segment portfolio groups. Daily and monthly frequency indices can be integrated with CRSPAccess97 Stock files to provide excess returns on the fly based on a number of criteria.

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 45 for more descriptive information about the methodologies of the CRSP index types.

Index Header contains the following types of information:

- Primary identifiers (Permanent Index Number, Permanent Index Group Number)
- Descriptive identifiers (Index Primary Link, Index Portfolio Number of Subset Series, Index Name, Index Group Name)
- Methodology Description Structure, containing Index Methodology Type Code, Index Primary Methodology Type, Index Secondary Methodology Type, Index Reweighting Type Flag, and Index Reweighting Timing Flag
- Index Exception Handling Flags Structure, containing Index Basic Exception Type Flag, Index New Issues Flag, Index Ineligible Issues Flag, Index Delisted Issues Flag, Index Missing Data Flag
- Index Subset Screening Structure and Partition Subset Screening Structure, each containing Universe Subset Type Code, Index Restriction Beginning Date, Index Restriction Ending Date, Valid Exchange Codes in the Universe, Valid Nasdaq Market Groups in the Universe, Valid When-Issued Securities in the Universe, Valid Incorporation of Securities in the Universe, and Share Code Screen Structure. The Share Code Screen Structure contains Share Code Groupings for Subsets, Valid First Digit of Share Code in the Universe, and Valid Second Digit of Share Code in the Universe.
- Portfolio Building Rules Structure, containing Index Basic Rule Types Code, Index Function Code for Buy Rules, Index Function Code for Sell Rules, Index Function Code for Generating Statistics, Index Statistic Grouping Code
- Related Assignment Information containing Index Basic Assignment Types Code, Permanent Index Identification Number of Associated Index, Portfolio Number in Associated Index, Calendar Identification Number of Assignment Calendar, Calendar Identification Number of Rebalancing Calendar, Calendar Identification Number of Calculations Calendar.

Index Header Object `[indhdr]`

Permanent Index Identification Number `(permno)`	Permanent Index Group Identification Number `(permco)`	Index Primary Link `(primflag)`	Portfolio Number if Subset Series `(portnum)`	Index Name `(indname)`
Index Group Name `(typename)`	Index Methodology Description Structure `(method)`	Index Exception Handling Flags `(flags)`	Index Subset Screening Structure `(partuniv)`	Share Code Screen Structure in a Partition Restriction `(shrcd)`
	Partition Subset Screening Structure `(induniv)`	Portfolio Building Rules Structure `(rules)`	Related Assignment Information `(assign)`	

Index Methodology Description Structure [method] Expansion

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

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

Universe Subset Types Code in a Restriction (partunivcode)	Restriction Beginning Date (begdt)	Restriction End Date (enddt)	Valid Exchange Codes in the Universe in a Restriction (wantexch)	Valid Nasdaq Market Groups in the Universe in a Restriction (wantnms)
	Valid When-Issued Securities in the Universe in a Restriction (wantwi)	Valid Incorporation of Securities in the 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

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

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)

Related Assignment Information [assign] Expansion

Index Basic Assignment Types Code (assigncode)	Permanent Index Identification Number 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 Arrays are a set of event array structures containing 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.

Each index rebalancing event structure contains the following information:

- Date ranges of the rebalancing period (Rebalancing Beginning Date and Rebalancing Ending Date)
- Breakpoint information (Statistic Maximum Identifier, Statistic Maximum in Period, Statistic Minimum Identifier, Statistic Minimum in Period, Statistic Median in Period, Statistic Average in Period)
- Statistics of the portfolio during the rebalancing period (Count Available as of Rebalancing, Count at End of Rebalancing Period, Maximum Count During Period, and Count Used as of Rebalancing)

Rebalancing History Array `[rebal[#][n]]`

Index Rebalancing Beginning Date (`rebbgdt`)	Index Rebalancing Ending Date (`rbenddt`)	Count Used as of Rebalancing (`usdcnt`)	Maximum Count During Period (`maxcnt`)	Count Available as of Rebalancing (`totcnt`)
Count at End of Rebalancing Period (`endcnt`)	Statistic Minimum Identifier (`minid`)	Statistic Maximum Identifier (`maxid`)	Statistic Minimum in Period (`minstat`)	Statistic Maximum in Period (`maxstat`)
	Statistic Median in Period (`medstat`)	Average Statistic 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 10 possible rebalancing arrays in Index Groups and 1 in 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 Arrays

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

Each Index List History array element contains the following information:

- Security identifier (Permanent Number of Securities in Index List)
- Date ranges when the security is a member of the index (First Date Included in List and Last Date Included in List), and
- Security characteristics within index (Index Subcategory Code and Weight of an Issue)

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

Permanent Number of Securities in Index List `(permno)`	First Date Included in List `(begdt)`	Last Date Included in a List `(enddt)`	Index Subcategory Code `(subind)`	Weight of an Issue `(weight)`

The variable Number of List Types contains the count of issue lists available for all indices in a set. There is 1 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 a list of result and summary time series for indices. They include the following variables:

- Index summary statistics (Portfolio Used Count, Portfolio Eligible Count, Portfolio Used Weight, and Portfolio Eligible Weight)
- Index returns (Index Total Returns, Index Capital Appreciation, and Index Income Return)
- Index levels (Index Level on Index Total Returns, Index Level on Index Capital Appreciation, and Index Level on Index Income Return).

Index Time Series Arrays

Array of Used Count Time Series (`usdcnt`)	Array of Total Count Time Series (`totcnt`)	Array of Used Value Time Series (`usdval`)	Array of Total Value Time Series (`totval`)	Array of Total Return Time Series (`tret`)
Array of Capital Appreciation Time Series (`aret`)	Array of Income Return Time Series (`iret`)	Array of Total Return Index Level Time Series (`tind`)	Array of Capital Appreciation Index Level Time Series (`aind`)	Array of Income Return Index Level Time Series (`iind`)

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.

CRSPAccess97 and SFA FORTRAN Index Structures

Calendar/Indices Arrays

Calendar/Indices arrays are described under Alternate SFA and FORTRAN Stock Structures. These include calendar date and index result arrays and are used for Market Indices data including the CRSP Indices for the S&P 500 Universe.

Calendar/Indices, Decile Returns, and Index Levels Arrays

Calendar/Indices, Decile Returns, and Index Level Arrays are a set of arrays for Stock File Indices when accessed using SFA character format files or CRSP FORTRAN access from the CRSP Indices and Portfolio Assignments File.

The arrays include the Calendar/Indices arrays plus the following arrays:

- Index Levels on the Market Indices in the Calendar/Indices. These are the Index Level Associated with the Return (Including Dividends) on a Value-Weighted Index, Index Level Associated with the Return (Excluding Dividends) on a Value-Weighted Index, Index Level Associated with the Return (Including Dividends) on an Equal-Weighted Index, and Index Level Associated with the Return (Excluding Dividends) on an Equal-Weighted Index.
- Return on Decile and Index Level Associated with the Return on Decile

CRSP Stock File Indices Array

Calendar Trading Date (CALDT)	Return (Including all Distributions) Value-Weighted Index (VWRETD)	Index Level Associated with the Return (Including all Distributions) on Value-Weighted Index (VWINDD)	Return (Excluding Dividends) on Value-Weighted Index (VWRETX)
Index Level Associated with the Return (Excluding Dividends) on Value-Weighted Index (VWINDX)	Return (Including all Distributions) on Equal-Weighted Index (EWRETD)	Index Level Associated with the Return (Including all Distributions) on Equal-Weighted Index (EWINDD)	Return (Excluding Dividends) on Equal-Weighted Index (EWRETX)
Index Level Associated with the Return (Excluding Dividends) on Equal-Weighted Index (EWINDX)	S&P 500 Composite Index Level (SPINDX) or Index Level on Nasdaq Composite (NCINDX)	S&P 500 Composite Index Return (SPRTRN) or Return on Nasdaq Composite Index (NCRTRN)	Return on Decile (DECRET)
Index Level Associated with the Return on Decile (DECIND)	Total Value of Market (TOTVAL)	Total Count of Market (TOTCNT)	

Cap-Based Reports Monthly History Arrays

Cap-Based Reports Monthly History Arrays are a set of arrays containing calendar and result data for Cap-Based Portfolios when accessed using SFA character format files or CRSP FORTRAN access from the CRSP Indices and Portfolio Assignments File or the Cap-Based Reports Historical Files.

The arrays include the following variables:

- Calendar Date
- Portfolio Name
- Portfolio Issue Count and Portfolio Weight
- Return on Portfolio and Index Level Associated with Total Return on Portfolio
- Capital Appreciation on Portfolio and Index Level Associated with the Capital Appreciation of Portfolio
- Return on Income Portfolio and Index Level Associated with the Income Return on Portfolio

CRSP Cap-Based Monthly History Array

Calendar Trading Date (CALDT)	Portfolio Sequence Number (PRTNAM)	Portfolio Issue Count (PRTCNT)	Portfolio Weight (PRTWGT)
Return on Portfolio (TOTRET)	Index Level Associated with Total Return on Portfolio (TOTIND)	Capital Appreciation on Portfolio (CAPRET)	Index Level Associated with the Capital Appreciation on Portfolio (CAPIND)
	Return on Income Portfolio (INCRET)	Index Level Associated with the Income Return on Portfolio (INCIND)	

Cap-Based Reports Rebalancing History Arrays

Cap-Based Reports Rebalancing History Arrays are a set of arrays containing calendar and rebalancing data for Cap-Based Portfolios when accessed using SFA character format files from the CRSP Indices and Portfolio Assignments File or the Cap-Based Reports Historical Files.

The arrays include the following variables:

- Year and Month of Quarter
- Portfolio Number of Decile
- Portfolio Company Count
- Capitalization of Smallest Company in Portfolio and Portfolio Smallest Company Name
- Capitalization of Largest Company in Portfolio and Portfolio Largest Company Name

CRSP Cap-Based Reports Rebalancing History Array

Year and Month of Quarter (YYYYMM)	Portfolio Number of Decile (PRTNO)	Portfolio Company Count (PRTCCT)	Capitalization of Smallest Company in Portfolio (MINCWT)
Portfolio Smallest Company Name (MINCNM)	Capitalization of Largest Company in Portfolio (MAXCWT)	Portfolio Largest Company Name (MAXCNM)	

CTI Indices Arrays

CTI Indices Arrays are a set of arrays containing calendar and result data for CRSP Treasury and Inflation data when accessed using SFA character format files or CRSP FORTRAN access using the CRSP Indices and Portfolio Assignments File.

The arrays include the following variables:

- Calendar Date
- Return on 30 Year Bonds and Index Level Associated with 30 Year Bonds
- Return on 20 Year Bonds and Index Level Associated with 20 Year Bonds
- Return on 10 Year Bonds and Index Level Associated with 10 Year Bonds
- Return on 7 Year Bonds and Index Level Associated with 7 Year Bonds
- Return on 5 Year Bonds and Index Level Associated with 5 Year Bonds
- Return on 2 Year Bonds and Index Level Associated with 2 Year Bonds
- Return on 1 Year Bonds and Index Level Associated with 1 Year Bonds
- Return on 90 Day Bonds and Index Level Associated with 90 Day Bonds
- Return on 30 Day Bonds and Index Level Associated with 30 Day Bonds
- Rate of Change in Consumer Price Index and Index Level Associated with Consumer Price Index

CRSP CTI Indices Array

Calendar Trading Date (CALDT)	Return on 30 Year Bonds (B30RET)	Index Level Associated with the 30 Year Bond Returns (B30IND)	Return on 20 Year Bonds (B20RET)
Index Level Associated with the 20 Year Bond Returns (B20IND)	Return on 10 Year Bonds (B10RET)	Index Level Associated with the 10 Year Bond Returns (B10IND)	Return on 7 Year Bonds (B7RET)
Index Level Associated with the 7 Year Bond Returns (B7IND)	Return on 5 Year Bonds (B5RET)	Index Level Associated with the 5 Year Bond Returns (B5IND)	Return on 2 Year Bonds (B2RET)
Index Level Associated with the 2 Year Bond Returns (B2IND)	Return on 1 Year Bonds (B1RET)	Index Level Associated with the 1 Year Bond Returns (B1IND)	Return on 90-Day Bills (T90RET)
Index Level Associated with the 90 Day Bill Returns (T90IND)	Return on 30-Day Bills (T30RET)	Index Level Associated with the 30 Day Bill Returns (T30IND)	Consumer Price Index Rate of Change (CPIRET)
	Index Level Associated with the Rate of Change in Consumer Price Index (CPIIND)		

Base CRSPAccess97 Data Structures

Time Series Objects

Time Series Objects are data structures used to store time series data in CRSPAccess97 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. The fields are:

— Data description information (Object Type Code, Array Type Code, Data Subcategory Type Code, Array Structure Size, and Maximum Number of Array Elements)

— Ranges of valid data (Begin Index of Valid Data, End Index of Valid Data)

— Associated Calendar information (Calendar Associated with Time Series, Calendar Time Period Description Code)

— Object Array with the list of observations for the time series. Each observation can be a complex CRSP data group such as the Portfolio Statistic and Assignment Time Series or a simple array such as Price or Bid/Ask Average.

CRSP Time Series Objects

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`)	Beginning Index of Valid Data (`beg`)	End Index of Valid Data (`end`)	Calendar Time Period Description Code (`caltype`)
	Calendar Associated with Time Series (`cal`)	Object Array (`arr`)	

Event Array Objects

Event Array Objects are data structures used to store event data in CRSPAccess97 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 list of event observations. The event times or effective time ranges are contained within the observations. The fields are:

— Data description information (Object Type Code, Array Type Code, Data Subcategory Type Code, Data Secondary Subcategory Type Code, Array Structure Size, and Maximum Number of Array Elements)

— Number of Array Elements

— Object Array with the list of event observations.

CRSP Array Objects

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`)	Number of Array Elements (`num`)	Data Secondary Subtype Code (`dummy`)	Object Array (`arr`)

Header Objects

Header Objects are data structures used to store header data in CRSPAccess97 databases. A CRSP header object contains information about the type of data stored and the actual header fields. The fields are:

— Data description information (Object Type Code, Array Type Code, Data Subcategory Type Code, and Array Structure Size)

— Object Array with header fields

CRSP Header Objects

Object Type Code (`objtype`)	Array Type Code (`arrtype`)	Data Subtype Code (`subtype`)	Array Structure Size (`size_of_array_width`)
	Object Array (`arr`)		

Calendar Objects

Calendar Objects are data structures used to store calendar data in CRSPAccess97 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. The fields are:

— Data description information (Object Type Code, Maximum Number of Array Elements, Calendar Type Availability Flag)

— Calendar description information (Calendar Name, Calendar Identification Number)

— Ranges of valid data (Number of Periods in Calendar)

— Calendar period arrays (Calendar Trading Date, Calendar Period Grouping Identifiers)

CRSP Calendar Objects

Object Type Code (`objtype`)	Calendar Identification Number (`calid`)	Maximum Number of Array Elements (`maxarr`)	Calendar Type Availability Flag (`loadflag`)
Number of Periods in Calendar (`ndays`)	Calendar Name(`name`)	Calendar Period Grouping Identifiers (`callist`)	Calendar Trading Date (`caldt`)

CHAPTER 3: CRSP INDEX METHODOLOGIES

OVERVIEW

This chapter describes the CRSP Index Methodologies.

INSIDE

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, and 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 Standard & Poor's 500 and Nasdaq Composite Index Data, and
- CRSP Stock File Capitalization Decile Indices or
  - CRSP Stock File Risk-Based Decile Indices
- CRSP Cap-Based Portfolios
- CRSP Indices for the S&P 500 Universe
- 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 151 and see "Appendix B: CRSP Terminology" on page 171 for more specific details about the calculations and definitions used.

3.1 Stock File Indices

The CRSP Stock File Indices are a set of Market Indices and Decile Portfolio Indices provided for five market groups of securities in daily, monthly, quarterly, and annual frequencies. The market groups of securities for which indices are calculated are the individual NYSE, AMEX, and Nasdaq markets, and the NYSE/AMEX and NYSE/AMEX/Nasdaq 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)	monthly, quarterly and annual	begins December 31, 1925
The New York Stock Exchange (NYSE)	daily series	begins July 2, 1962
The American Stock Exchange (AMEX)	all series	begins July 2, 1962
The Nasdaq Stock Market (NASDAQ)	all series	begins December 14, 1972

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. Quarterly and annual series are only available in SFA format in the Indices and Portfolio Assignments File.

CRSP Market Indices

There are two indices created for each market group, an Equal-Weighted Index and a Value-Weighted Index. Each index contains index returns with and without dividends, weights and counts.

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

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

The NYSE/AMEX/Nasdaq Market Indices are available for Daily and Monthly Stock Files. Other exchange combinations are available in the Indices and Portfolio Assignments File.

Published Standard and Poor's 500 and Nasdaq Composite Index Data

CRSP provides the levels of the S&P 500 Composite Index and Nasdaq Composite Index and calculates returns on the levels. The S&P 500 Composite Index levels are collected from publicly available sources such as the Dow Jones News Service, the *Wall Street Journal*, or the *Standard and Poor's Statistical Service*. Nasdaq Composite Index levels are provided by Nasdaq.

The S&P 500 Composite Index level is a value-weighted index created by Standard and Poor's. 500 securities have been included in the index since March 1957. Before that it was called the S&P 90 Index and included 90 securities. The index does not include dividends. See the CRSP Indices for the S&P 500 Universe for indices based on total returns on the same universe of stocks.

The Nasdaq Composite Index level is a value-weighted index created by the Nasdaq Stock Market. The index does not include dividends.

Published Standard and Poor's 500 and Nasdaq Composite Index Data are provided with daily and monthly CRSPAccess97 Stock Files.

CRSP Stock File Capitalization Decile Indices

The CRSP Stock File Capitalization Decile Indices use a different methodology and universe than the CRSP Cap-Based Indices. CRSP Stock File Capitalization Decile Indices are calculated for each of the Stock File Indices market groups. In these Market Segment Indices 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 is 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 one contains the securities with the highest statistics, and portfolio ten contains the securities with the lowest statistics.

Once securities are assigned to portfolios an equal-weighted index with dividends included 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 then 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.

Cap-Based Portfolios

CRSP Cap-Based Portfolios are monthly series of capitalization-based market segments using a different methodology and universe than the CRSP Stock File Capitalization Decile Indices. Cap-Based Portfolio indices are monthly series based on portfolios rebalanced quarterly. Monthly history and quarterly rebalancing history data are provided.

The universe includes all common stocks listed on the NYSE, AMEX, and The Nasdaq National Market excluding:

- Unit Investment Trusts, Closed-End Funds, Real Estate Investment Trusts, Americus Trusts, Foreign Stocks, and American Depository Receipts.

All eligible companies listed on the NYSE are ranked by market capitalization and then split into ten equally populated groups, or deciles. The capitalization of the largest company in each decile serves as the breakpoint for that decile. When multiple issues of a company trade, the sum of the issue capitalizations is used for the company capitalization so that the company is counted only once. The portfolios are reformed every quarter using the price and shares at the end of the previous quarter.

The largest companies are placed in portfolio 1, the smallest in portfolio 10. In addition to the 10 individual portfolios, CRSP produces a single return number for the portfolio formed by combining: Deciles 1 and 2 to create CRSP 1-2; Deciles 3, 4 and 5 to create CRSP 3-5; Deciles 1 through 5 to create CRSP 1-5, Deciles 6, 7 and 8 to create CRSP 6-8; Deciles 9 and 10 to create CRSP 9-10; Deciles 6 through 10 to create CRSP 6-10; Deciles 1 through 10 form the market portfolio.

There are three series based on exchange:

NYSE only.

NYSE and AMEX. AMEX data are added beginning July, 1962

NYSE + AMEX + The Nasdaq National Market. The Nasdaq National Market data are added beginning April, 1982.

Breakpoints for all three series are based exclusively on companies with issues traded on the NYSE. In the second and third series, non-NYSE companies are assigned to appropriate portfolios according to their capitalization in relation to the decile breakpoints.

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 the last price is a month-end price, that 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 National Market, 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.

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 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 data beginning July 2, 1962, and monthly data beginning December 25, 1925 are provided. The published S&P 500® index and returns are also included for comparison. 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.

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.

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 Treasury and Inflation Indices

CRSP US Treasury and Inflation Series (CTI): The CRSP US Treasury and Inflation Series (CTI) Files are provided on a monthly frequency. The series contains returns adapted from the CRSP US Government Bond 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 the 30, 20, 10, 7, 5, 2, and 1 year fixed term indices, a valid issue that best represents each term is chosen at the end of each month for each of the seven fixed terms and held through the next month. Valid issues are at least six months from the target maturity date and are fully taxable. The selection process consists of selecting a representative bond from each of the fixed term groups by filtering available issues on the basis of their characteristics. First, a non-callable, non-flower bond that is closest to the target maturity of its group and fully taxable is found. If more than one security meets these criteria, the one most recently issued is used. If there is no other suitable issue, a second pass is made where flower bonds are accepted. Due to the unavailability of consistently suitable issues for maturities before 1950, the series contain missing values before the starting dates given in the individual variable descriptions below.

The issue selected in the 30-day series is the Treasury Bill closest to but with not less than 30 days to maturity. The 90-day series uses a 90-day target. For these two series, where bills were not available certificates and, in a few cases, notes were used. The selection amongst alternatives was somewhat subjective in early periods. The issue with the maturity closest to target was sometimes rejected because the quotes were suspicious. In no case was an issue used which did not mature on its next coupon payment date. Also excluded were issues with bid quotations implying negative yields. This resulted in some very short nominally 90-day maturities prior to 1942. Similarly, scarcity of available issues results in some very long nominal one month issues being used prior to 1937. The range of maturities of both series after 1942 is within a few days of the targets. Users may wish to restrict their usage to this period.

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 CALDT was selected based on its length to maturity as of the CALDT. The returns contained in these series are calculated under the assumption that the relevant issue is bought one month prior to the CALDT and sold on the CALDT.

The issue chosen for the 90 and 30 day Treasury Bill series on a given CALDT was selected based on its length to maturity as of the month immediately prior to the CALDT. The 90 and 30-day series returns were calculated on the basis of buying the relevant issue one month prior to the CALDT and selling it on the CALDT. For example, a 90-day bill return is calculated between a date approximately 90 days prior to the bill's maturity and the CALDT, 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 CALDT, which is a date one month later. In cases where the CALDT 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.

3.2 CRSP Index Series and Groups

The following table contains a list of all CRSP Index Series by Permanent Index Identification Number.

Index Series	indno	Daily	Monthly	Product Availability
CRSP NYSE Value-Weighted Market Index	1000000	Yes	Yes	IX
CRSP NYSE Equal-Weighted Market Index	1000001	Yes	Yes	IX
CRSP NYSE Market Capitalization Decile 1	1000002	Yes	Yes	IX
CRSP NYSE Market Capitalization Decile 2	1000003	Yes	Yes	IX
CRSP NYSE Market Capitalization Decile 3	1000004	Yes	Yes	IX
CRSP NYSE Market Capitalization Decile 4	1000005	Yes	Yes	IX
CRSP NYSE Market Capitalization Decile 5	1000006	Yes	Yes	IX
CRSP NYSE Market Capitalization Decile 6	1000007	Yes	Yes	IX
CRSP NYSE Market Capitalization Decile 7	1000008	Yes	Yes	IX
CRSP NYSE Market Capitalization Decile 8	1000009	Yes	Yes	IX
CRSP NYSE Market Capitalization Decile 9	1000010	Yes	Yes	IX
CRSP NYSE Market Capitalization Decile 10	1000011	Yes	Yes	IX
CRSP AMEX Value-Weighted Market Index	1000020	Yes	Yes	IX
CRSP AMEX Equal-Weighted Market Index	1000021	Yes	Yes	IX
CRSP AMEX Market Capitalization Decile 1	1000022	Yes	Yes	IX
CRSP AMEX Market Capitalization Decile 2	1000023	Yes	Yes	IX
CRSP AMEX Market Capitalization Decile 3	1000024	Yes	Yes	IX
CRSP AMEX Market Capitalization Decile 4	1000025	Yes	Yes	IX
CRSP AMEX Market Capitalization Decile 5	1000026	Yes	Yes	IX
CRSP AMEX Market Capitalization Decile 6	1000027	Yes	Yes	IX
CRSP AMEX Market Capitalization Decile 7	1000028	Yes	Yes	IX
CRSP AMEX Market Capitalization Decile 8	1000029	Yes	Yes	IX
CRSP AMEX Market Capitalization Decile 9	1000030	Yes	Yes	IX
CRSP AMEX Market Capitalization Decile 10	1000031	Yes	Yes	IX
CRSP NYSE/AMEX Value-Weighted Market Index	1000040	Yes	Yes	IX
CRSP NYSE/AMEX Equal-Weighted Market Index	1000041	Yes	Yes	IX
CRSP NYSE/AMEX Market Capitalization Decile 1	1000042	Yes	Yes	IX
CRSP NYSE/AMEX Market Capitalization Decile 2	1000043	Yes	Yes	IX
CRSP NYSE/AMEX Market Capitalization Decile 3	1000044	Yes	Yes	IX
CRSP NYSE/AMEX Market Capitalization Decile 4	1000045	Yes	Yes	IX
CRSP NYSE/AMEX Market Capitalization Decile 5	1000046	Yes	Yes	IX
CRSP NYSE/AMEX Market Capitalization Decile 6	1000047	Yes	Yes	IX
CRSP NYSE/AMEX Market Capitalization Decile 7	1000048	Yes	Yes	IX
CRSP NYSE/AMEX Market Capitalization Decile 8	1000049	Yes	Yes	IX
CRSP NYSE/AMEX Market Capitalization Decile 9	1000050	Yes	Yes	IX
CRSP NYSE/AMEX Market Capitalization Decile 10	1000051	Yes	Yes	IX
CRSP Nasdaq Value-Weighted Market Index	1000060	Yes	Yes	IX
CRSP Nasdaq Equal-Weighted Market Index	1000061	Yes	Yes	IX
CRSP Nasdaq Market Capitalization Decile 1	1000062	Yes	Yes	IX
CRSP Nasdaq Market Capitalization Decile 2	1000063	Yes	Yes	IX
CRSP Nasdaq Market Capitalization Decile 3	1000064	Yes	Yes	IX
CRSP Nasdaq Market Capitalization Decile 4	1000065	Yes	Yes	IX
CRSP Nasdaq Market Capitalization Decile 5	1000066	Yes	Yes	IX
CRSP Nasdaq Market Capitalization Decile 6	1000067	Yes	Yes	IX
CRSP Nasdaq Market Capitalization Decile 7	1000068	Yes	Yes	IX
CRSP Nasdaq Market Capitalization Decile 8	1000069	Yes	Yes	IX
CRSP Nasdaq Market Capitalization Decile 9	1000070	Yes	Yes	IX
CRSP Nasdaq Market Capitalization Decile 10	1000071	Yes	Yes	IX
CRSP NYSE/AMEX/Nasdaq Value-Weighted Market Index	1000080	Yes	Yes	DA, MA, IX
CRSP NYSE/AMEX/Nasdaq Equal-Weighted Market Index	1000081	Yes	Yes	DA, MA, IX
CRSP NYSE/AMEX/Nasdaq Market Capitalization Decile 1	1000082	Yes	Yes	IX
CRSP NYSE/AMEX/Nasdaq Market Capitalization Decile 2	1000083	Yes	Yes	IX
CRSP NYSE/AMEX/Nasdaq Market Capitalization Decile 3	1000084	Yes	Yes	IX
CRSP NYSE/AMEX/Nasdaq Market Capitalization Decile 4	1000085	Yes	Yes	IX
CRSP NYSE/AMEX/Nasdaq Market Capitalization Decile 5	1000086	Yes	Yes	IX
CRSP NYSE/AMEX/Nasdaq Market Capitalization Decile 6	1000087	Yes	Yes	IX
CRSP NYSE/AMEX/Nasdaq Market Capitalization Decile 7	1000088	Yes	Yes	IX
CRSP NYSE/AMEX/Nasdaq Market Capitalization Decile 8	1000089	Yes	Yes	IX
CRSP NYSE/AMEX/Nasdaq Market Capitalization Decile 9	1000090	Yes	Yes	IX

CRSP NYSE/AMEX/Nasdaq Market Capitalization Decile 10	1000091	Yes	Yes	IX
CRSP NYSE/AMEX Beta Decile 1	1000102	Yes	-	IX
CRSP NYSE/AMEX Beta Decile 2	1000103	Yes	-	IX
CRSP NYSE/AMEX Beta Decile 3	1000104	Yes	-	IX
CRSP NYSE/AMEX Beta Decile 4	1000105	Yes	-	IX
CRSP NYSE/AMEX Beta Decile 5	1000106	Yes	-	IX
CRSP NYSE/AMEX Beta Decile 6	1000107	Yes	-	IX
CRSP NYSE/AMEX Beta Decile 7	1000108	Yes	-	IX
CRSP NYSE/AMEX Beta Decile 8	1000109	Yes	-	IX
CRSP NYSE/AMEX Beta Decile 9	1000110	Yes	-	IX
CRSP NYSE/AMEX Beta Decile 10	1000111	Yes	-	IX
CRSP NYSE/AMEX Standard Deviation Decile 1	1000122	Yes	-	IX
CRSP NYSE/AMEX Standard Deviation Decile 2	1000123	Yes	-	IX
CRSP NYSE/AMEX Standard Deviation Decile 3	1000124	Yes	-	IX
CRSP NYSE/AMEX Standard Deviation Decile 4	1000125	Yes	-	IX
CRSP NYSE/AMEX Standard Deviation Decile 5	1000126	Yes	-	IX
CRSP NYSE/AMEX Standard Deviation Decile 6	1000127	Yes	-	IX
CRSP NYSE/AMEX Standard Deviation Decile 7	1000128	Yes	-	IX
CRSP NYSE/AMEX Standard Deviation Decile 8	1000129	Yes	-	IX
CRSP NYSE/AMEX Standard Deviation Decile 9	1000130	Yes	-	IX
CRSP NYSE/AMEX Standard Deviation Decile 10	1000131	Yes	-	IX
CRSP Nasdaq Beta Decile 1	1000142	Yes	-	IX
CRSP Nasdaq Beta Decile 2	1000143	Yes	-	IX
CRSP Nasdaq Beta Decile 3	1000144	Yes	-	IX
CRSP Nasdaq Beta Decile 4	1000145	Yes	-	IX
CRSP Nasdaq Beta Decile 5	1000146	Yes	-	IX
CRSP Nasdaq Beta Decile 6	1000147	Yes	-	IX
CRSP Nasdaq Beta Decile 7	1000148	Yes	-	IX
CRSP Nasdaq Beta Decile 8	1000149	Yes	-	IX
CRSP Nasdaq Beta Decile 9	1000150	Yes	-	IX
CRSP Nasdaq Beta Decile 10	1000151	Yes	-	IX
CRSP Nasdaq Standard Deviation Decile 1	1000162	Yes	-	IX
CRSP Nasdaq Standard Deviation Decile 2	1000163	Yes	-	IX
CRSP Nasdaq Standard Deviation Decile 3	1000164	Yes	-	IX
CRSP Nasdaq Standard Deviation Decile 4	1000165	Yes	-	IX
CRSP Nasdaq Standard Deviation Decile 5	1000166	Yes	-	IX
CRSP Nasdaq Standard Deviation Decile 6	1000167	Yes	-	IX
CRSP Nasdaq Standard Deviation Decile 7	1000168	Yes	-	IX
CRSP Nasdaq Standard Deviation Decile 8	1000169	Yes	-	IX
CRSP Nasdaq Standard Deviation Decile 9	1000170	Yes	-	IX
CRSP Nasdaq Standard Deviation Decile 10	1000171	Yes	-	IX
CRSP NYSE Cap-Based Portfolio 3-5	1000311	-	Yes	IX
CRSP NYSE Cap-Based Portfolio 6-8	1000312	-	Yes	IX
CRSP NYSE Cap-Based Portfolio 9-10	1000313	-	Yes	IX
CRSP NYSE Cap-Based Portfolio 1-5	1000314	-	Yes	IX
CRSP NYSE Cap-Based Portfolio 6-10	1000315	-	Yes	IX
CRSP NYSE Cap-Based Portfolio Market	1000316	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 1	1000320	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 2	1000321	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 3	1000322	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 4	1000323	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 5	1000324	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 6	1000325	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 7	1000326	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 8	1000327	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 9	1000328	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 10	1000329	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 1-2	1000330	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 3-5	1000331	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 6-8	1000332	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 9-10	1000333	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 1-5	1000334	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio 6-10	1000335	-	Yes	IX
CRSP NYSE/AMEX Cap-Based Portfolio Market	1000336	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 1	1000340	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 2	1000341	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 3	1000342	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 4	1000343	-	Yes	IX

CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 5	1000344	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 6	1000345	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 7	1000346	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 8	1000347	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 9	1000348	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 10	1000349	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 1-2	1000350	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 3-5	1000351	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 6-8	1000352	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 9-10	1000353	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 1-5	1000354	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio 6-10	1000355	-	Yes	IX
CRSP NYSE/AMEX/Nasdaq National Market Cap-Based Portfolio Market	1000356	-	Yes	IX
CRSP Value-Weighted Index of the S&P 500 Universe	1000500	Yes	Yes	IX
CRSP Equal-Weighted Index of the S&P 500 Universe	1000501	Yes	Yes	IX
S&P 500 Composite	1000502	Yes	Yes	DA, MA, IX
Nasdaq Composite	1000503	Yes	Yes	DA, MA, IX
CRSP 30-Year Bond Returns	1000700	-	Yes	IX
CRSP 20-Year Bond Returns	1000701	-	Yes	IX
CRSP 10-Year Bond Returns	1000702	-	Yes	IX
CRSP 7-Year Bond Returns	1000703	-	Yes	IX
CRSP 5-Year Bond Returns	1000704	-	Yes	IX
CRSP 2-Year Bond Returns	1000705	-	Yes	IX
CRSP 1-Year Bond Returns	1000706	-	Yes	IX
CRSP 90-Day Bill Returns	1000707	-	Yes	IX
CRSP 30-Day Bill Returns	1000708	-	Yes	IX
Consumer Price Index	1000709	-	Yes	IX

The following table contains a list of all CRSP Index Groups by Permanent Index Identification Number.

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

3.3 Portfolio Types Defined by CRSP

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

CHAPTER 4: DATA DEFINITIONS

OVERVIEW

This chapter contains stock and indices data definitions in alphabetical order, organized by name.

INSIDE

CHAPTER 4: DATA DEFINITIONS

This section has been modified to facilitate user access to the CRSP data. It contains the CRSP stock and indices variables organized by name, in alphabetical order. Each includes a detailed definition and a usage table for the variable. The usage table contains the following items:

Primary Concept	Always contains the name of the primary data groups described in the data structure section, and may contain references to calculations or index methodologies.		
***ts_print* Daily Usage**	the data item/`subno` used to access the variable from a daily CRSPAccess97 database with the *ts_print* utility. See the CRSPAccess97 Database Format - Utilities Guide for usage of *ts_print*	blank cell	
***ts_print* Monthly Usage**	same as daily, but for use with a monthly CRSPAccess97 database.	***stk_print* Option(s)**	Options that can be used to access the variable from the *stk_print* utility. See the CRSPAccess97 Database Format - Utilities Guide for usage of *stk_print*
C Object or C Object Type	The name of the CRSPAccess97 data object needed to access the variable using C, or the name of the base CRSPAccess97 data structure(s) containing these elements	**FORTRAN Common Block**	The name of the common block containing the variable using FORTRAN77 using either the CRSPAccess97 or the SFA Database format
C Array or **C Variable Equivalent**	The name of the CRSPAccess97 array used to access the variable using C, or The C variable equivalent to use if the specific item is a FORTRAN-only data variable.	**FORTRAN Array or** **FORTRAN Variable Equivalent**	The name of the array used to access the variable using FORTRAN77, or the FORTRAN variable equivalent to use if the specific item is a C-only data variable.
C Element or C Structure **C Equivalent Usage**	The mnemonic name of the structure element or structure within the array used to access the variable using C, or the C equivalent usage that can be used to pull the equivalent value if the data item is a FORTRAN-only variable.	**FORTRAN Element or FORTRAN Variable or** **FORTRAN Equivalent Usage**	The mnemonic name of the structure element parameter within the array used to access the variable using FORTRAN, the FORTRAN variable name, or the FORTRAN usage that can be used to pull the equivalent value if the data item is a C-only variable.
Database Format(s)	The database format needed to access the variable. Possible values are: CA97-CRSPAccess97 format database and SFA-SFA format database.	**Data 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 data to utilize, `IND`-Indices Data, and `IND‡`-Index data item requires stock data to utilize.

See the CRSPAccess97 Database Format - Programmers Guide for programming usage of a CRSPAccess97 database and the CRSP SFA Database Format Guide for FORTRAN programming of an SFA database .

The *ts_print* and *stk_print* items included in this section are at present, primary direct mappings. For utility program usage and available data items, see the CRSPAccess97 Database Format - Utilities Guide.

Most FORTRAN variables are available in both CRSPAccess97 and SFA database formats, whereas C variables are only available in CRSPAccess97. A cross-reference for usage is not yet included.

Variables that are used in other variable's defintions are italicized for easy recognition and cross reference.

Tables for C and FORTRAN are included in the **Database Structure Section**, 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.

The mnemonic `indno` is used in the usage table to indicate a specific *Permanent Index Identification Number* associated with the variable

Acquiring PERMCO

Acquiring PERMCO is the *PERMCO* of another company linked to a distribution. If the *Acquiring PERMNO* is nonzero 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	Distribution Event Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/di`
C Object	`dists_arr`	FORTRAN Common Block	`/INFO/`
C Array	`dists[]`	FORTRAN Array	`DISTS(,)`
C Element	`accomp`	FORTRAN Element	`ACCOMP`
Database Format(s)	`CA97`	Data Type(s)	`STK`

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 code on the same *Ex-Distribution Date*, they are numbered in the Acquiring PERMNO field to distinguish them. Data in this field is incomplete prior to 1985.

Primary Concept	Distribution Event Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/di`
C Object	`dists_arr`	FORTRAN Common Block	`/INFO/`
C Array	`dists[]`	FORTRAN Array	`DISTS(,)`
C Element	`acperm`	FORTRAN Element	`ACPERM`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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 a trade that occurred during the delisting month.

Primary Concept	Delisting History Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/de`
C Object	`delist_arr`	FORTRAN Common Block	`/INFO/`
C Array	`delist`	FORTRAN Array	`RDELIS(,)`
C Element	`dlamt`	FORTRAN Element	`DLAMT`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Array Structure Size

Arrray Structure Size is the number of bytes needed in each structure element for this array type in a CRSPAccess97 object structure.

Primary Concept	Base CRSPAccess97Data Structure		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_*	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	size_of_array_ width	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK, IND

Array Type Code

Array Type Code is an integer code which defines the type of data in a CRSPAccess97 object structure array. It can define a basic data type or a CRSP-defined structure.

Primary Concept	Base CRSPAccess97Data Structure		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_*	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	arrtype	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK, IND

Ask

Ask is the closing Nasdaq ask on a trading date. It is only available for issues trading on the Nasdaq Stock Market[SM] during time periods when *Ask or High Price* can contain the high price. Ask is reported for all securities listed on The Nasdaq National Market since November 1, 1982, and all Nasdaq securities since June 15, 1992.

The close of the day is 4:00 PM Eastern time. Since July 1980, Nasdaq has used the inside quotation as the closing bid and ask. The inside quotation is the highest bid and lowest ask.

Due to source limitations, data is missing for 15 Nasdaq National Market securities in December 1982, and all Nasdaq National Market securities in February 1986.

monthly: Ask is the closing ask on the last trading date of each month. This item was originally only available the Daily Stock Supplemental Nasdaq Data. It is incorporated as a standard data variable in all of the CRSPAccess97 databases. The Monthly SFA Database does not support this item, except as an optional time series choice in CRSPAccess97 to SFA conversion programs.

Primary Concept	Supplemental Nasdaq Data Arrays		
ts_print Daily Usage	ask/0		
ts_print Monthly Usage	mask/0	*stk_print* Option(s)	/pa
C Object	ask_ts	FORTRAN Common Block	/NMSDAT/
C Array	ask[]	FORTRAN Array	NMSASK(,)
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA - daily only	Data Type(s)	STK

Ask or High Price

daily: Ask or High Price is the highest closing price or closing ask price, when the closing price is not available. The field is set to zero if no Ask or High Price is available.

Daily trading prices for the Nasdaq National Market securities were first reported November 1, 1982. Daily trading prices for The Nasdaq Small Cap Market were first reported June 15, 1992. Ask or High for Nasdaq securities is always an ask before these dates.

monthly: Monthly files contain the highest daily Closing 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. The field is set to zero if no Ask or bid/ask average was available during the month.

Primary Concept	Price, Volume, and Return Arrays		
ts_print Daily Usage	`askhi/0`		
ts_print Monthly Usage	`maskhi/0`	*stk_print* Option(s)	`/ph`
C Object	`askhi_ts`	FORTRAN Common Block	`/DDATA/`
C Array	`askhi[]`	FORTRAN Array	`ASKHI()`
C Element	`n/a`	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Average Statistic in Period

Average Statistic in Period is the average statistical value in a portfolio at the beginning of a rebalancing period of a market segment index.

Primary Concept	Index Rebalancing History		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`rebal_arr[]`	FORTRAN Common Block	`n/a`
C Array	`rebal[][]`	FORTRAN Array	`n/a`
C Element	`avgstat`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`IND`

Begin Index of Beta Excess Return Data

daily: Begin Index of Beta Excess Return Data is the index of the first calendar period with valid *Beta Excess Returns* when *Beta Excess Returns* data is loaded into the *Optional Time Series 2* data slot. If no *Optional Time Series 2* data is available, the begin index is set to zero. The *Calendar Trading Date* at this index is the date of the first calendar period with data of this type for this security.

Primary Concept	Header Identification and Date Range Variables, Optional Data Arrays		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`n/a`	FORTRAN Common Block	`/HEADER/`
C Array	`n/a`	FORTRAN Array	`n/a`
C Element	`n/a`	FORTRAN Variable	`BEGBXS`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK‡`

Begin Index of Nasdaq Data

Begin Index of Nasdaq Data is the index of the first calendar period with valid Supplemental Nasdaq Data. If no supplemental data is available, the begin index is set to zero. The *Calendar Trading Date* at this index is the date of the first calendar period with data of this type for this security.

Primary Concept	Supplemental Nasdaq Header and Date Range Variables, Supplemental Nasdaq Data Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/NMSHDR/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	BEGNMS
Database Format(s)	CA97, SFA	Data Type(s)	STK

Begin Index of Optional Time Series 1 Data

Begin Index of Optional Time Series 1 Data is the index of the first calendar period with valid data loaded in the *Optional Time Series 1* data slot. If no *Optional Time Series 1* data is available, the begin index is set to zero. The *Calendar Trading Date* at this index is the date of the first calendar period with data of this type for this security.

Primary Concept	Header Identification and Date Range Variables, Optional Data Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	BEGSXS
Database Format(s)	CA97, SFA	Data Type(s)	STK*

Begin Index of Optional Time Series 2 Data

Begin Index of Optional Time Series 2 Data is the index of the first calendar period with valid data loaded in the *Optional Time Series 2* data slot. If no *Optional Time Series 2* data is available, the begin index is set to zero. The *Calendar Trading Date* at this index is the date of the first calendar period with data of this type for this security.

Primary Concept	Header Identification and Date Range Variables, Optional Data Arrays.		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	BEGPR2
Database Format(s)	CA97, SFA	Data Type(s)	STK*

Begin Index of Portfolio Data

Begin Index of Portfolio Data is the index containing the first portfolio statistic, values, or portfolio assignments available for a specific security. Begin Index of Portfolio Data + 1924 equals the actual year in YYYY format of data corresponding to this index If no portfolio data is available, the begin index is set to zero. The *Calendar Trading Date* at this index is the date of the first calendar period with data of this type for this security. This index applies to all three portfolio types in the Portfolio Assignment Array and in the Portfolio Statistic Array.

Primary Concept	Header Identification and Date Range Variables Portfolio Assignments Array, Portfoliko Statistics Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	BEGYR
Database Format(s)	CA97, SFA	Data Type(s)	STK*

Begin Index of Price Data

Begin Index of Price Data is the index of the first calendar period with valid price data in the file for a security. The *Calendar Trading Date* at this index is the date of the first calendar period with data of this type for this security.

Primary Concept	Header Identification and Date Range Variables, Price, Return and Volume Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	BEGPRC
Database Format(s)	CA97, SFA	Data Type(s)	STK

Begin Index of Return Data

Begin Index of Return Data is the index of the first calendar period with valid return data for a security. If no data of this type is 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 for this security.

Primary Concept	Header Identification and Date Range Variables, Price, Return and Volume Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	BEGRET
Database Format(s)	CA97, SFA	Data Type(s)	STK

Begin Index of Return without Dividends Data

Begin Index of Return without Dividends Data is the index of the first calendar period with valid *Return without Dividends* data when *Return without Dividends* data is loaded into the *Optional Time Series 1* data slot. If no *Optional Time Series 1* data is available, the begin index is set to zero. The *Calendar Trading Date* at this index is the date of the first calendar period with data of this type for this security.

Primary Concept	Header Identification and Date Range Variables Optional Data Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	BEGRTX
Database Format(s)	CA97, SFA	Data Type(s)	STK

Begin Index of Secondary Price Data

Begin Index of Secondary Price Data is the index of the first calendar period with valid *Bid or Low Price* or *Ask or High Price* data for a security. If no data of this type is 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 for this security.

Primary Concept	Header Information and Date Range Variables Price, Volume, and Return Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	BEGSP
Database Format(s)	CA97, SFA	Data Type(s)	STK

Begin Index of Spread between Bid and Ask

monthly: Begin Index of Spread between Bid and Ask Data is the index of the first calendar period with valid *Spread between Bid and Ask* data, when *Spread between Bid and Ask* data is loaded into the *Optional Time Series 2* data slot. If no *Optional Time Series 2* data is available, the begin index is set to zero. The *Calendar Trading Date* at this index is the date of the first calendar period with data of this type for this security.

.

Primary Concept	Header Identification and Date Range Variables Optional Data Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	BEGPR2
Database Format(s)	CA97, SFA	Data Type(s)	STK

Begin Index of Standard Deviation Excess Return Data

daily: Begin Index of Standard Deviation Excess Return Data is the index of the first calendar period with valid *Standard Deviation Excess Return* data when *Standard Deviation Excess Return* data is loaded into the *Optional Time Series 1* data slot. If no *Optional Time Series 1* data is available, the begin index is set to zero. The *Calendar Trading Date* at this index is the date of the first calendar period with data of this type for this security.

Primary Concept	Header Identification and Date Range Variables, Optional Data Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	BEGSXS
Database Format(s)	CA97, SFA	Data Type(s)	STK‡

Begin Index of Stock Data

Begin Index of Stock Data is the index of the first calendar period with valid data for a security. If no data of this type is 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 for this security.

Primary Concept	Header Identification and Date Range Variables.		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	BEGDAT
Database Format(s)	CA97, SFA	Data Type(s)	STK

Begin Index of Valid Data

Begin Index of Valid Data is the index of the first calendar period with valid data for a security. If no data of this type is 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 for this security.

Primary Concept	Base CRSPAccess97 Data Structures		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_TIMESERIES	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	beg	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK, IND

Begin Index of Volume Data

Begin Index of Volume Data is the index of the first calendar period with valid volume data for a security. If no data of this type is 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 for this security.

Primary Concept	Header Identification and Date Range Variables Price, Volume, and Return Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	BEGVOL
Database Format(s)	CA97, SFA	Data Type(s)	STK

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. When used with *Calendar Trading Date*, it is the first date for a security with data in this field.

Primary Concept	Header Identification and Summary Data		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hh, /hr, /hrl
C Object	n/a	FORTRAN Common Block	n/a
C Array	header	FORTRAN Array	n/a
C Element	begdt	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK

Beta Excess Return

Daily: Beta Excess Return denotes the excess return of a specific issue less the average return of all issues in its beta portfolio each trading date. It is important to note that for NYSE and AMEX data, the beta is computed by using trade-only returns data. (i.e. excluding returns on bid-ask averages). Nasdaq data betas are calculated from bid-ask averages. A missing return due to a portfolio assignment of zero is set to -44.0. Beta Excess Return is only available in a daily database when the CRSP US Stock, Treasury Indices and Portfolio Assignments Database is available. See *Optional Time Series 2*.

Primary Concept	Optional Data Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/ADATA/
C Array	n/a	FORTRAN Array	BXRET()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	STK‡

Bid

Bid is the closing bid on a trading date. It is only available for issues trading on The Nasdaq Stock Market^{SM} during time periods when *Bid or Low Price* can contain the low price. Bid is reported for all securities listed on The Nasdaq National Market since November 1, 1982, and all Nasdaq securities since June 15, 1992.

The close of the day is 4:00 PM Eastern time. Since July 1980, Nasdaq has used the inside quotation as the closing bid and ask. The inside quotation is the highest bid and lowest ask.

Due to source limitations, data is missing for 15 Nasdaq National Market securities in December, 1982, and all The Nasdaq National Market securities in February, 1986.

monthly: Monthly files contain the highest daily closing Bid during the month.

Primary Concept	Supplemental Nasdaq Data Arrays		
ts_print Daily Usage	`bid/0`		
ts_print Monthly Usage	`mbid/0`	*stk_print* Option(s)	`/pb`
C Object	`bid_ts`	FORTRAN Common Block	`/NMSDAT/`
C Array	`bid[]`	FORTRAN Array	`NMSBID()`
C Element	`n/a`	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Bid or Low Price

daily: Bid or Low Price is the lowest closing price or closing bid price, 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 Small Cap Market were first reported June 15, 1992. Bid or Low for Nasdaq securities is always an ask before these dates.

monthly: Monthly files contain the lowest daily Closing 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. The field is set to zero if no Ask or bid/ask average is available.

Primary Concept	Price, Volume and Return Arrays		
ts_print Daily Usage	`bidlo/0`		
ts_print Monthly Usage	`mbidlo/0`	*stk_print* Option(s)	`/pl, /dd`
C Object	`bidlo_ts`	FORTRAN Common Block	`/DDATA/`
C Array	`bidlo[]`	FORTRAN Array	`BIDLO()`
C Element	`n/a`	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Calendar Associated with Time Series

Calendar Associated with a Time Series is a pointer in a CRSP time series object to the associated CRSPAccess97 calendar structure needed to assign the time to time series array data.

Primary Concept	Calendars		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_TIMESERIES	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	cal	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK, IND

Calendar Identification Number

Calendar Identification Number is an integer code assigned by CRSP to trading calendars. A Calendar Name Calendar Identification Number identify each calendar. The calendars supported in CRSPAccess97 databases are:

Calendars	Calendar Identification Number	Calendar Name	Beginning Date
daily	100	Daily Trading Calendar	19620702
monthly	101	Month-end Trading Calendar	19251231
annual	300	Annual Trading Calendar	19251231
quarterly	310	Quarterly Trading Calendar	19251231
weekly	500	Weekly Trading Calendar	19620706

Primary Concept	Base CRSPAccess97 Data Structures		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_CAL	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	calid	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK, IND

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 period numbers 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 assignment calendar uses the same calendars listed in *Calendar Identification Number.*

Primary Concept	Index Header Calendars		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	assigncal	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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 periodicity numbers 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	Index Header Calendars		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	calccal	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	rebalcal	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Calendar Name

Calendar Name is a text description of a calendar.

Primary Concept	Base CRSPAccess97 Data Structures		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_CAL	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	name	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK, IND

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 for the calendar periods. The values stored for each period in Calendar Period Grouping Identifier for the current calendars are:

Calendar	Date Format
daily	YYYYWW
monthly	YYYYMM
annual	YYYY
quarterly	YYYYMM
weekly	YYYYWW

Where WW is 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	Base CRSPAccess97 Data Structures		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_CAL	FORTRAN Common Block	n/a
C Array	callist[]	FORTRAN Array	n/a
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK, IND

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 time series. This code currently always equals 2, indicating that calendar time periods are identified by the last trading date in the period using *Calendar Trading Date*.

Primary Concept	Base CRSPAccess97 Data Structrures		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_TIMESERIES	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	caltype	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK, IND

Calendar Trading Date

Calendar Trading Date contains the list of trading dates. Each date represents the last date in a calendar period, in YYYYMMDD (year, month, day) format. These dates begin in 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	Base CRSPAccess97 Data Structures		
ts_print Daily Usage	caldt/0		
ts_print Monthly Usage	mcaldt/0	*stk_print* Option(s)	all time series use /dt for range restriction
C Object Type	CRSP_CAL	FORTRAN Common Block	/CAL/
C Array	caldt[]	FORTRAN Array	CALDT()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	STK, IND

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	Base CRSPAccess97 Data Structures		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_CAL	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	loadflag	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK, IND

Capital Appreciation on Portfolio

Capital Appreciation on Portfolio is the capital appreciation on the selected portfolio on the selected *Calendar Trading Date* for a CRSP Cap-Based Portfolio.

Primary Concept	Cap-Based Reports Monthly History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	retx/0 (cap-based index)	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAPBAS/
C Array	n/a	FORTRAN Array	CAPRET(,)
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Capitalization of Largest Company in Portfolio

Capitalization of Largest Company In Portfolio is the capitalization of the largest company included in a CRSP Cap-Based portfolio at the beginning of each Quarter.

Primary Concept	Cap-Based Reports Rebalancing History		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/REBAL/
C Array	n/a	FORTRAN Array	MAXCWT(,)
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Capitalization of Smallest Company in Portfolio

Capitalization of Smallest Company in Portfolio is the capitalization of the smallest company included in a CRSP Cap-Based portfolio at the beginning of the Quarter.

Primary Concept	Cap-Based Reports Rebalancing History		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/REBAL/
C Array	n/a	FORTRAN Array	MINCWT(,)
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Company Name

Company Name is the name of the company at the time of its name history. CRSP allocates a 32 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 32 characters, CRSP furnishes its own abbreviations.

Primary Concept	Name History Array		
ts_print Daily Usage	`comnam/0`		
ts_print Monthly Usage	`mcomnam/0`	*stk_print* Option(s)	`/n`
C Object	`names_arr`	FORTRAN Common Block	`/INFO/`
C Array	`names`	FORTRAN Array	`CNAMES(,)`
C Element	`comnam`	FORTRAN Element	`COMNAM` or `COMPNM` (function)
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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 32 characters long.

Primary Concept	Header Identification and Summary Data		
ts_print Daily Usage	`comnam/2`		
ts_print Monthly Usage	`mcomnam/2`	*stk_print* Option(s)	`/hh, /hr, /hrl`
C Object	`header_row`	FORTRAN Common Block	`n/a`
C Array	`header`	FORTRAN Array	`n/a`
C Element	`hcomnam`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`STK`

Consumer Price Index Rate of Change

The Consumer Price Index Rate of Change for all urban consumers, not seasonally adjusted (`CPI-U NSA`), measures inflation, which is the rate of change of prices of consumer goods. The inflation measures are constructed by the US Department of Labor, Bureau of Labor Statistics.

Primary Concept	CTI Indices Arrays		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`mcapret/0 (indno 1000709)`	*stk_print* Option(s)	`n/a`
C Object	`n/a`	FORTRAN Common Block	`/CTIS/`
C Array	`n/a`	FORTRAN Array	`CPIRET()`
C Element	`n/a`	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`IND`

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.

Primary Concept	Index Rebalancing History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	rebal_arr[]	FORTRAN Common Block	n/a
C Array	rebal[][]	FORTRAN Array	n/a
C Element	endcnt	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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.

Primary Concept	Index Rebalancing History		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	rebal_arr[]	FORTRAN Common Block	n/a
C Array	rebal[][]	FORTRAN Array	n/a
C Element	totcnt	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Count of Securities Used

Count of Securities Used 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.

Primary Concept	Calendar/Indices Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAL/
C Array	n/a	FORTRAN Array	USDCNT()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

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.

Primary Concept	Index Rebalancing History		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	rebal_arr	FORTRAN Common Block	n/a
C Array	rebal	FORTRAN Array	n/a
C Element	usdcnt	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

CUSIP

CUSIP refers to the CUSIP identifier valid during the date range of the name structure. All non-blank CUSIPS 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 CUSIPs 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 Identifier, Header" on page 72.

Primary Concept	Name History Array		
ts_print Daily Usage	`ncusip/0`		
ts_print Monthly Usage	`mncusip/0`	*stk_print* Option(s)	`/n`
C Object	`names_arr`	FORTRAN Common Block	`/INFO/`
C Array	`names[]`	FORTRAN Array	`CNAMES(,)`
C Element	`ncusip`	FORTRAN Element	`NCUSIP` or `CUS()P` (Function)
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

CUSIP Identifier, Header

CUSIP Identifier, Header is the latest eight-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 CUSIPs are 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 Header CUSIPs, which are assigned by CRSP. One, ***99*9*, (containing a 9 in the 4^{th}, 5^{th} and 7^{th} 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 7^{th} character position) represents a CRSP-assigned cusip with a real issuer number but a dummy issue number. For example:

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

CUSIP Identifier, Header *(Con't)*

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 Issuer (6 characters) Issue (2 characters) per 8-character identifier system set by the CUSIP Numbering System. 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. **See the current CUSIP Directory for more information. See Appendix A.1 for a list of CINS country codes**

Primary Concept	Header Identification and Summary Variables, Header Identification and Date Range Variables		
ts_print Daily Usage	`cusip/0`		
ts_print Monthly Usage	`mcusip/0`	*stk_print* Option(s)	`/hh, /hr, /hrl`
C Object	`header_row`	FORTRAN Common Block	`/HEADER/`
C Array	`header`	FORTRAN Array	`n/a`
C Element	`hcusip`	FORTRAN Variable	`CUSIP`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Data Secondary Subtype Code

Data Secondary Subtype Code is an integer code further defining the *Data Subtype Code*. It is set to 0 if unused.

Primary Concept	Base CRSPAccess97 Data Structures		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object Type	`CRSP_ARRAY`	FORTRAN Common Block	`n/a`
C Array	`n/a`	FORTRAN Array	`n/a`
C Element	`dummy`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`STK, IND`

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 0 if unused. The Data Subtype Code for a portfolio assignments and statistics time series is set to the *Permanent Index Identification Number* of the index group with portfolio results for the market segment portfolio type.

Primary Concept	Base CRSPAccess97 Data Structures		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object Type	`CRSP_*`	FORTRAN Common Block	`n/a`
C Array	`n/a`	FORTRAN Array	`n/a`
C Element	`subtype`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`STK, IND`

Delisting Code

Delisting Code is a three-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.

Delisting Codes

Primary 1st Digit of Code	Category
1	still trading or halted but not yet delisted
2	merger
3	exchange
4	liquidation
5	delisted by NYSE, AMEX, or Nasdaq
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 details of delisting events. Additional delisting codes, specific to various delisting categories, have been created to indicate if an issue is closed to further research, or if the issue is pending further research. See "6.3 Delisting Codes" on page 166 for details of delisting coding schemes.

Primary Concept	Delisting History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/de
C Object	delist_arr	FORTRAN Common Block	/INFO/
C Array	delist[]	FORTRAN Array	DELIST(,)
C Element	dlstcd	FORTRAN Element	DLSTCD
Database Format(s)	CA97, SFA	Data Type(s)	STK

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	Header Identification and Summary Data		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hh, /hr, /hrl
C Object	header_row	FORTRAN Common Block	n/a
C Array	header	FORTRAN Array	n/a
C Element	dlstcd	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK

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 trading date of the file. Delisting date is never missing.

monthly: Delisting Date is not necessarily a month-end trading date and may not be found in the *Calendar Trading Date* array.

Primary Concept	Delisting History Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`delist_arr`	FORTRAN Common Block	`/INFO/`
C Array	`delist[]`	FORTRAN Array	`DELIST(,)`
C Element	`dlstdt`	FORTRAN Element	`DLSTDT`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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 *Delisting Date*. 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	Delisting History Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/de`
C Object	`delist_arr`	FORTRAN Common Block	`/INFO/`
C Array	`delist[]`	FORTRAN Array	`DELIST(,)`
C Element	`nextdt`	FORTRAN Element	`NEXTDT`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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	Delisting History Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/de`
C Object	`delist_arr`	FORTRAN Common Block	`/INFO/`
C Array	`delist[]`	FORTRAN Array	`DELIST(,)`
C Element	`dlpdt`	FORTRAN Element	`DLPDT`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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	Delisting History Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/de`
C Object	`delist_arr`	FORTRAN Common Block	`/INFO/`
C Array	`delist`	FORTRAN Array	`RDELIS(,)`
C Element	`dlprc`	FORTRAN Element	`DLPRC`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Delisting Return

Delisting Return is the return of a security after it has delisted from the New York, American, or NASDAQ stock exchange. 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 was found. (This only applies 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.

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	security is still active.
-99.0	security trades on a new exchange after delisting, but CRSP currently has no sources to gather price information.

Any issue that is pending further research, has delisting codes 470 and 480, only, with a missing return code set to -55.

Primary Concept	Delisting History Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/de`
C Object	`delist_arr`	FORTRAN Common Block	`/INFO/`
C Array	`delist[]`	FORTRAN Array	`RDELIS(,)`
C Element	`dlret`	FORTRAN Element	`DLRET`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Delisting Return without Dividends

Delisting Return Without Dividends is the return of a security after it has delisted from the New York, American, or NASDAQ stock exchange. 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 Calculation Section (Page 152) for calculation and missing values.

Primary Concept	Delisting History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/de
C Object	delist_arr	FORTRAN Common Block	/INFO/
C Array	delist[]	FORTRAN Array	RDELIS(,)
C Element	dlretx	FORTRAN Element	DLRETX
Database Format(s)	CA97, SFA	Data Type(s)	STK

Distribution Code

CRSP describes company distributions and corporate actions in the distribution history with a four-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 161 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	Distribution Event Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/di
C Object	dists_arr	FORTRAN Common Block	/INFO/
C Array	dists[]	FORTRAN Array	DISTS(,)
C Element	distcd	FORTRAN Element	DISTCD
Database Format(s)	CA97, SFA	Data Type(s)	STK

Distribution Declaration Date

Distribution Declaration Date is the date (in YYYYMMDD format) on which the board of directors declared a distribution. If a declaration can not 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	Distribution Event Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/di
C Object	dists_arr	FORTRAN Common Block	/INFO/
C Array	dists[]	FORTRAN Array	DISTS(,)
C Element	dclrdt	FORTRAN Element	DCLRDT
Database Format(s)	CA97, SFA	Data Type(s)	STK

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. When the distribution is paid in shares of a trading security, the Dividend Cash Amount is set to 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.

Primary Concept	Distribution Event Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/di
C Object	dists_arr	FORTRAN Common Block	/INFO/
C Array	dists[]	FORTRAN Array	RDISTS(,)
C Element	divamt	FORTRAN Element	DIVAMT
Database Format(s)	CA97, SFA	Data Type(s)	STK

End Index of Beta Excess Return Data

daily: End Index of Beta Excess Return Data is the index of the last calendar period with valid *Beta Excess Returns* when *Beta Excess Returns* data is loaded into the *Optional Time Series 2* data slot. If no *Optional Time Series 2* data is available, the end index 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	Header Identification and Date Range Variables, Optional Data Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	ENDBXS
Database Format(s)	CA97, SFA	Data Type(s)	STK‡

End Index of Nasdaq Data

End Index of Nasdaq Data is the index of the last calendar period with valid Supplemental Nasdaq Data. If no supplemental data is available, the end index 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	Supplemental Nasdaq Header and Date Range Variables, Supplemental Nasdaq Data Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/NMSHDR/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	ENDNMS
Database Format(s)	CA97, SFA	Data Type(s)	STK

End Index of Optional Time Series 1 Data

End Index of Optional Time Series 1 Data is the index of the last calendar period with valid data loaded in the *Optional Time Series 1* data slot. If no *Optional Time Series 1* data is available, the end index 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	Header Identification Information and Date Range Variables, Optional Data Arrays.		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	ENDSXS
Database Format(s)	CA97, SFA	Data Type(s)	STK

End Index of Optional Time Series 2 Data

End Index of Optional Time Series 2 Data is the index of the last calendar period with valid data loaded in the *Optional Time Series 2* data slot. If no *Optional Time Series 2* data is available, the end index 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	Header Identification Information and Date Range Variables, Optional Data Arrays.		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	ENDBXS
Database Format(s)	CA97, SFA	Data Type(s)	STK

End Index of Portfolio Data

End Index of Portfolio Data is the index containing the last portfolio statistic, values, or portfolio assignments available for a specific security. End Index of Portfolio Data + 1924 equals the actual year in YYYY format of data corresponding to this index If no portfolio data is available, the end index 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. This index applies to all three portfolio types in the Portfolio Assignment Array and in the Portfolio Statistic Array.

Primary Concept	Stock Header and Date Range Variables Arrays, Portfolio Assignments		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	ENDYR
Database Format(s)	CA97, SFA	Data Type(s)	STK, IND*

End Index of Price Data

End Index of Price Data is the index of the last calendar period with valid price data in the file for a security. 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	Stock Header and Date Range Variables, Price, Return and Volume Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	ENDPRC
Database Format(s)	CA97, SFA	Data Type(s)	STK

End Index of Return Data

End Index of Return Data is the index of the last calendar period with valid return data for a security. If no data of this type is 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	Stock Header and Date Range Variables, Price, Return and Volume Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	ENDRET
Database Format(s)	CA97, SFA	Data Type(s)	STK

End Index of Return without Dividends Data

End Index of Return without Dividends Data is the index of the last calendar period with valid *Return without Dividends* data when *Return without Dividends* data is loaded into the *Optional Time Series 1* data slot. If no *Optional Time Series 1* data is available, the end index 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	Header Identification and Date Range Variables		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	ENDRTX
Database Format(s)	CA97, SFA	Data Type(s)	STK

End Index of Secondary Price Data

End Index of Secondary Price Data is the index of the last calendar period with valid *Bid or Low Price* or *Ask or High Price* data for a security. If no data of this type is 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	Header Information and Date Range Variables Price, Volume, and Return Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	ENDSP
Database Format(s)	CA97, SFA	Data Type(s)	STK

End Index of Spread between Bid and Ask

monthly: End Index of Spread between Bid and Ask Data is the index of the last calendar period with valid *Spread between Bid and Ask* data, when *Spread between Bid and Ask* data is loaded into the *Optional Time Series 2* data slot. If no *Optional Time Series 2* data is available, the end index 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	Header Identification and Date Range Variables Optional Data Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	ENDPR2
Database Format(s)	CA97, SFA	Data Type(s)	STK

End Index of Standard Deviation Excess Return Data

daily: End Index of Standard Deviation Excess Return Data is the index of the last calendar period with valid *Standard Deviation Excess Return* data when *Standard Deviation Excess Return* data is loaded into the *Optional Time Series 1* data slot. If no *Optional Time Series 1* data is available, the end index 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	Header Identification and Date Range Variables, Optional Data Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	ENDSXS
Database Format(s)	CA97, SFA	Data Type(s)	STK*

End Index of Stock Data

End Index of Stock Data is the index of the last calendar period with valid data for a security. If no data of this type is 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	Header Identification and Date Range Variables		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	ENDDAT
Database Format(s)	CA97, SFA	Data Type(s)	STK

End Index of Valid Data

End Index of Valid Data is the index of the last calendar period with valid data for a security. If no data of this type is 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	Base CRSPAccess97 Data Structures		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_TIMESERIES	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	end	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK, IND

End Index of Volume Data

End Index of Volume Data is the index of the last calendar period with valid volume data for a security. If no data of this type is 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	Header Identification and Date Range Variables Price, Volume, and Return Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	ENDVOL
Database Format(s)	CA97, SFA	Data Type(s)	STK

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. When used with *Calendar Trading Date*, it is the last date for a security with data in this field.

Primary Concept	Header Identification and Summary Data		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hh, /hr, /hrl
C Object	n/a	FORTRAN Common Block	n/a
C Array	header	FORTRAN Array	n/a
C Element	enddt	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK

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	Distribution Event Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/di
C Object	dists_arr	FORTRAN Common Block	/INFO/
C Array	dists[]	FORTRAN Array	DISTS(,)
C Element	exdt	FORTRAN Element	EXDT
Database Format(s)	CA97, SFA	Data Type(s)	STK

Exchange Code

Exchange Code is an integer code indicating the exchange on which a security is listed. Primary exchange codes are respectively 1, 2, and 3 for the New York Stock Exchange, the American Stock Exchange, and The Nasdaq Stock Market℠. 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, 3, and 33) identifies when-issued trading, such as during a reorganization. The following table contains a list of primary exchange codes included in the name history. See "North American Security Exchange & Indices Codes" on page 160 for a complete list of exchange codes.

Exchange Codes

Code	Definition
-2	Suspended by the NYSE or AMEX
-1	Halted by the NYSE or AMEX
0	Not listed on exchange of current file
1	New York Stock Exchange
2	American Stock Exchange
3	The Nasdaq Stock Market℠
31	When-issued trading on the New York Stock Exchange
32	When-issued trading on the American Stock Exchange
33	When-issued trading on The Nasdaq Stock Market℠

Primary Concept	Name History Array		
ts_print Daily Usage	`exchcd/0`		
ts_print Monthly Usage	`mexchcd/0`	*stk_print* Option(s)	`/n`
C Object	`names_arr`	FORTRAN Common Block	`/INFO/`
C Array	`names[]`	FORTRAN Array	`NAMES(,)`
C Element	`exchcd`	FORTRAN Element	`EXCHCD`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Exchange Code, Header

Exchange Code, Header displays the Exchange Code of the primary exchange where a security was last listed. Valid Exchange Code, Header values are 1, 2, or 3, which correspond to the NYSE, AMEX, and Nasdaq respectively. *Other exchange codes are not included in the Exchange Code, Header field.*

Primary Concept	Header Identification and Summary Data Header Identification and Date Range Variables		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/hh, /hr, /hrl`
C Object	`header_row`	FORTRAN Common Block	`/HEADER/`
C Array	`header`	FORTRAN Array	`n/a`
C Element	`hexcd`	FORTRAN Variable	`HEXCD`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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 Ordinary cash dividends or partial liquidating payments, Factor to Adjust Price is set to zero;

2 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.

In other less common distribution events, spin-offs 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 days after *(P(t))*, CRSP substitutes that price for *(P(t)).*

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 negative one.

monthly: Since only month-end prices are available on the monthly Stock Files, it is possible to use the Factor to Adjust Price and the *Dividend Cash Amount* to compute the approximate price of the underlying security on the ex-distribution date. Distributions with Factor to Adjust Price are in effect reinvested on the *Ex-Distribution Date* and not at the end of the month.

Primary Concept	Distribution History Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/di`
C Object	`dists_arr`	FORTRAN Common Block	`/INFO/`
C Array	`dists[]`	FORTRAN Array	`RDISTS(,)`
C Element	`facpr`	FORTRAN Element	`FACPR`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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 three types of distributions where this is the case. See *Factor to Adjust Price* (Page 86) for these cases and how they are handled.

For spin-offs, Factor to Adjust Shares Outstanding is set to zero. For rights issues the 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 the same as for stock splits.

Primary Concept	Distribution History Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/di`
C Object	`dists_arr`	FORTRAN Common Block	`/INFO/`
C Array	`dists[]`	FORTRAN Array	`RDISTS(,)`
C Element	`facshr`	FORTRAN Element	`FACSHR`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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.

Primary Concept	Index List History Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`list_arr[]`	FORTRAN Common Block	`n/a`
C Array	`list[][]`	FORTRAN Array	`n/a`
C Element	`begdt`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`IND`

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 156.

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 return codes specify the reason a return is missing.

Missing Return Codes

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	outside the range of the security's price range
-99.0	missing return due to missing price at time *t*; usually due to suspension in trading or tradin on unknown exchange.

Primary Concept	Price, Volume and Return Arrays		
ts_print Daily Usage	`ret/0`		
ts_print Monthly Usage	`mret/0`	*stk_print* Option(s)	`/pr, /dd, /dr, /dx`
C Object	`ret_ts`	FORTRAN Common Block	`/DDATA/`
C Array	`ret[]`	FORTRAN Array	`RET()`
C Element	`n/a`	FORTRAN Variable	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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	Index Header		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`indhdr_row`	FORTRAN Common Block	`n/a`
C Array	`indhdr`	FORTRAN Array	`n/a`
C Element	`assigncode`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`IND`

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	flagcode	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	rulecode	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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: *Basic Exception Type Code*, *Index New Issues Flag*, *Index Ineligible Issues Flag*, *Return of Delisted Issues Flag*, and *Index Missing Data Flag*.

Primary Concept	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C structure	flags	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header Index Rebalancing History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	buyfnct	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	statfnct	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	sellfnct	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	groupname	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	delflag	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Index Level Associated with Return on 1 Year Bonds

Index Level Associated with Return on 1-Year Bonds is the index level for 1-year bonds on the selected *Calendar Trading Date*. Index levels are set based on an initial investment of $100.00 on December 29, 1972. (*Index Identification Number* 1000706)

Primary Concept	CTI Indices Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	tind (for indno 1000706)	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	B1IND()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with Return on 10 Year Bonds

Index Level Associated with Return on 10-Year Bonds is the index level for 10-year bonds on the selected *Calendar Trading Date*. Index levels are set based on an initial investment of $100.00 on December 29, 1972. (*Index Identification Number* 1000702)

Primary Concept	CTI Indices Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	B10IND()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with Return on 2 Year Bonds

Index Level Associated with Return on 2-Year Bonds is the index level for 2-year bonds on the selected *Calendar Trading Date*. Index levels are set based on an initial investment of $100.00 on December 29, 1972. (*Index Identification Number* 1000705)

Primary Concept	CTI Indices Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	B2IND()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with Return on 20 Year Bonds

Index Level Associated with Return on 20 Year Bonds is the index level for 20 year bonds on the selected *Calendar Trading Date*. (*Index Identification Number* 1000701)

Primary Concept	CTI Indices Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	B20IND()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with Return on 30-Day Bills

Index Level Associated with Return on 30-Day Bills is the index level for 30-day bills on the selected *Calendar Trading Date*. Index levels are set based on an initial investment of $100.00 on December 29, 1972. (*Index Identification Number* 1000708)

Primary Concept	CTI Indices Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	T30IND()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with Return on 30 Year Bonds

Index Level Associated with Return on 30 Year Bonds is the index level for 30 year bonds on the selected *Calendar Trading Date*. (*Index Identification Number* 1000700)

Primary Concept	CTI Indices Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	B30IND()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with Return on 5 Year Bonds

Index Level Associated with Return on 5-Year Bonds is the index level for 5-year bonds on the selected *Calendar Trading Date.* Index levels are set based on an initial investment of $100.00 on December 29, 1972. (*Index Identification Number* 1000704)

Primary Concept	CTI Indices Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	B5IND()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with Return on 7 Year Bonds

Index Level Associated with Return on 7-Year Bonds is the index level for 7-year bonds on the selected *Calendar Trading Date.* Index levels are set based on an initial investment of $100.00 on December 29, 1972. (*Index Identification Number* 1000703)

Primary Concept	CTI Indices Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	B7IND()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with Return on 90-Day Bills

Index Level Associated with Return on 90-Day Bills is the index level for 90-day bills on the selected *Calendar Trading Date.* Index levels are set based on an initial investment of $100.00 on December 29, 1972. (*Index Identification Number* 1000707)

Primary Concept	CTI Indices Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	T90IND()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with Capital Appreciation on Portfolio

Index Level Associated with Capital Appreciation on Portfolio is the index level, associated with capital appreciation for the selected CRSP Cap-Based portfolio on the selected *Calendar Trading Date*. (*Index Identification Numbers* 1000340-1000357)

Primary Concept	Capital Appreciation		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAPBAS/
C Array	n/a	FORTRAN Array	CAPIND()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with Income Return on Portfolio

Index Level Associated with Income Return on Portfolio is the index level, associated with income returns on the selected Cap-Based portfolio on the selected *Calendar Trading Date*. (*Permanent Index Identification Numbers* 1000340-1000357)

Primary Concept	Income Return		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAPBAS/
C Array	n/a	FORTRAN Array	INCIND()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with the Rate of Change in Consumer Price Index

Index level Associated with the Rate of Change of Consumer Price Index is the Consumer Price Index for all urban consumers, not seasonally adjusted (CPI-U NSA). It is used to measure inflation as the rate of change of prices of consumer goods. The inflation measures are constructed by the US Departement of Labor, Bureau of Labor Statistics. (*Index Identification Number* 1000709)

Primary Concept	CTI Indices Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	CPIIND()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with Return on Decile

Index Level Associated with Return on Decile is the index level of the CRSP stock file index decile portfolios on the selected *Calendar Trading Date*.

Primary Concept	CRSP Stock File Indices		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/DECILE/
C Array	n/a	FORTRAN Array	DECIND()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with the Return (Excluding Dividends) on Equal-Weighted Index

Index Level Associated with the Return (Excluding Dividends) on Equal-Weighted Index is the index level associated with the return on the CRSP stock file equal-weighted market index, excluding dividends, on the selected *Calendar Trading Date*.

Primary Concept	CRSP Stock File Indices		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/DECILE/
C Array	n/a	FORTRAN Array	EWINDX()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with the Return (Including all Distributions) on Equal-Weighted Index

Index Level Assocated with the Return (Including all Distributions) on Equal-Weighted Index is the index level associated with the CRSP stock file equal-weighted market index, excluding dividends, on the selected *Calendar Trading Date*.

Primary Concept	CRSP Stock File Indices		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/DECILE/
C Array	n/a	FORTRAN Array	EWINDD()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with the Return (Excluding Dividends) on Value-Weighted Index

Index Level Associated with the Return (Excluding Dividends) on Value-Weighted Index is the index level associated with the CRSP stock file value-weighted market index, excluding dividends, on the selected *Calendar Trading Date*.

Primary Concept	CRSP Stock File Indices		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAL/
C Array	n/a	FORTRAN Array	VWINDX()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with the Return (Including all Distributions) on Value-Weighted Index

Index Level Associated with the Return (Including all Distributions) on Value-Weighted Index is the index level associated with the return on the CRSP stock file value-weighted market index, including all distributions, on the selected *Calendar Trading Date*.

Primary Concept	CRSP Stock File Indices		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAL/
C Array	n/a	FORTRAN Array	VWINDD()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level Associated with Total Return on Portfolio

Index Level Associated with Total Return on Portfolio is the index level associated with total returns on a CRSP Cap-Based portfolio on the selected *Calendar Trading Date*.

Primary Concept	Cap-Based Portfolios		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAPBAS/
C Array	n/a	FORTRAN Array	TOTIND(,)
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Index Level on Nasdaq Composite

Index Level on Nasdaq Composite is the level of the publicly reported Nasdaq Composite Index at the end of the trading period. This data is collected from Nasdaq and does not include dividends. The index indicates the change in price of the component securities.

Primary Concept	CRSP Stock File Indices		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAL/
C Array	n/a	FORTRAN Array	NCINDX()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	STK, IND

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 45. 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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	methcode	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Structure	method	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	missflag	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Index Name

Index Name is the name of the index or portfolio. The index names are listed in the *Permanent Index Identification Number* table. See “3.2 CRSP Index Series and Groups” on page 51.

Primary Concept	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	indname	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`indhdr_row`	FORTRAN Common Block	`n/a`
C Array	`indhdr`	FORTRAN Array	`n/a`
C Element	`addflag`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`IND`

Index Primary Link

Index Primary Link is the *Permanent Index Group Identification Number* 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	Index Header		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`indhdr_row`	FORTRAN Common Block	`n/a`
C Array	`indhdr`	FORTRAN Array	`n/a`
C Element	`primflag`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`IND`

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	primtype	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	CA97

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	Index Rebalancing History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	rebal_ts[]	FORTRAN Common Block	n/a
C Array	rebal[][]	FORTRAN Array	n/a
C Element	rbbegdt	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Rebalancing History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	rebal_ts[]	FORTRAN Common Block	n/a
C Array	rebal[][]	FORTRAN Array	n/a
C Element	rbenddt	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		n/a
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	wgtflag	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header Portfolio Statistics & Assignments Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	wgttype	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	subtype	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	groupflag	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index List History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	list_arr[]	FORTRAN Common Block	n/a
C Array	list[][]	FORTRAN Array	n/a
C Element	subind	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Structure	partuniv	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Last Date Included in a 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.

Primary Concept	Index List History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	list_arr[]	FORTRAN Common Block	n/a
C Array	list[][]	FORTRAN Array	n/a
C Element	enddt	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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, or the *Delisting Date* if 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 will be exactly one such name structure for any date between the first *Name Effective Date* and the *Delisting Date*.

Primary Concept	Name History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/n (with /fs)
C Object	names_arr	FORTRAN Common Block	n/a
C Array	names[]	FORTRAN Array	n/a
C Element	nameenddt	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK

Market Maker Count

Market Maker Count is the number of registered market makers for an issue trading on the Nasdaq Stock Market. This contains a 0 if there are no registered 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	Nasdaq Information Array		
ts_print Daily Usage	`mmcnt/0`		
ts_print Monthly Usage	`mmmcnt/0`	*stk_print* Option(s)	`/q`
C Object	`nasdin_arr`	FORTRAN Common Block	`/INFO/`
C Array	`nasdin[]`	FORTRAN Array	`NASDIN(,)`
C Element	`mmcnt`	FORTRAN Element	`MMCNT`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Market Value of Securities Used

Market Value of Securities Used is the total market value, in $1000's, of all securities that are used in a value-weighted index on the selected *Calendar Trading Date*. To be used for value-weighting, a security cannot be an ADR and must have valid prices on the current and previous trading days.

Primary Concept	Calendars/Indices, Market Indices		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`n/a`	FORTRAN Common Block	`/CAL/`
C Array	`n/a`	FORTRAN Array	`USDVAL`
C Element	`n/a`	FORTRAN Variable	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK*, IND`

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	Index Rebalancing Arrays		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`rebal_arr[]`	FORTRAN Common Block	`n/a`
C Array	`rebal[][]`	FORTRAN Array	`n/a`
C Element	`maxcnt`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`IND`

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	Base CRSPAccess97 Data Structures		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_TIMESERIES, CRSP_CAL, CRSP_ARRAY	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	maxarr	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK, IND

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 *Share Class* 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	Name History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/n
C Object	names_arr	FORTRAN Common Block	/INFO/
C Array	names[]	FORTRAN Array	NAMES(,)
C Element	namedt	FORTRAN Element	NAMEDT
Database Format(s)	CA97, SFA	Data Type(s)	STK

NASD Index Code

NASD Index Code is a one-digit integer code indicating the issue's classification within NASD's internal business description categories. This field is not available after April, 1998. The following codes are used in the file.

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

Primary Concept	Nasdaq Information Arra		
ts_print Daily Usage	nsdinx/0		
ts_print Monthly Usage	mnsdinx/0	*stk_print* Option(s)	/q
C Object	nasdin_arr	FORTRAN Common Block	/INFO/
C Array	nasdin[]	FORTRAN Array	NASDIN(,)
C Element	nsdinx	FORTRAN Element	NSDINX
Database Format(s)	CA97, SFA	Data Type(s)	STK

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	Header Identification and Summary Data		
ts_print Daily Usage	`compno/0`		
ts_print Monthly Usage	`mcompno/0`	*stk_print* Option(s)	`/hh, /hr, /hrl`
C Object	`header_row`	FORTRAN Common Block	`n/a`
C Array	`header`	FORTRAN Array	`n/a`
C Element	`compno`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`STK`

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	Header Identification and Summary Data Header Identification and Date Range Variables		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/hh, /hr, /hrl`
C Object	`header_row`	FORTRAN Common Block	`/HEADER/`
C Array	`header`	FORTRAN Array	`n/a`
C Element	`issuno`	FORTRAN Variable	`ISSUNO`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Nasdaq National Market Indicator

Nasdaq National Market Indicator is a one-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 Small Cap Securities. As of June 15, 1992 transaction data became available.

Code	Description
0	unknown or unavailable
1	The Nasdaq Small Cap Market before June 15, 1992
2	The Nasdaq National Market
3	The Nasdaq Small Cap Market after June 15, 1992

Primary Concept	Nasdaq Information Array		
ts_print Daily Usage	`nmsind/0`		
ts_print Monthly Usage	`mnmsind/0`	*stk_print* Option(s)	`/q`
C Object	`nasdin_arr`	FORTRAN Common Block	`/INFO/`
C Array	`nasdin[]`	FORTRAN Array	`NASDIN(,)`
C Element	`nmsind`	FORTRAN Element	`NMSIND`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Nasdaq Traits Code

Nasdaq Traits Code is a one-digit integer code describing the trading status of an issue listed on the Nasdaq Stock Market. This code is not available after April, 1998 and is set to 0 for all issues after that time. 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	Nasdaq Information Array		
ts_print Daily Usage	`trtscd/0`		
ts_print Monthly Usage	`mtrtscd/0`	*stk_print* Option(s)	`/q`
C Object	`nasdin_arr`	FORTRAN Common Block	`/INFO/`
C Array	`nasdin[]`	FORTRAN Array	`NASDIN(,)`
C Element	`trtscd`	FORTRAN Element	`TRTSCD`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Nasdaq Traits Date

Nasdaq Traits 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	Nasdaq Information Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/q
C Object	nasdin_arr	FORTRAN Common Block	/INFO/
C Array	nasdin[]	FORTRAN Array	NASDIN(,)
C Element	trtsdt	FORTRAN Element	TRTSDT
Database Format(s)	CA97, SFA	Data Type(s)	STK

Nasdaq Traits End Date

Nasdaq Traits End Date is the last date, in YYYYMMDD format, that 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	Nasdaq Information Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/q
C Object	nasdin_arr	FORTRAN Common Block	n/a
C Array	nasdin[]	FORTRAN Array	n/a
C Element	trtsenddt	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK

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 payments not primarily in the stock of the company. New PERMCO if zero is 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	Delisting History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/de
C Object	delist_arr	FORTRAN Common Block	/INFO/
C Array	delist[]	FORTRAN Array	DELIST(,)
C Element	nwcomp	FORTRAN Element	NWCOMP
Database Format(s)	CA97	Data Type(s)	STK

New PERMNO

New PERMNO is an integer link 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 0 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	Delisting History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	delist_arr	FORTRAN Common Block	/INFO/
C Array	delist[]	FORTRAN Array	DELIST(,)
C Element	nwperm	FORTRAN Element	NWPERM
Database Format(s)	CA97, SFA	Data Type(s)	STK

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	Base CRSPAccess97 Data Structure		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_ARRAY	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	num	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK, IND

Number of Delisting Structures

The Number of Delisting Structures is the number of delisting events in the delisting history array for a security. The Number of Delisting Structures is always equal to one.

Primary Concept	Header Identification and Date Range Variables Delisting History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	NUMDEL
Database Format(s)	CA97, SFA	Data Type(s)	STK

Number of Distribution Structures

Number of Distribution Structures is the number of distribution events available in the distribution event array for a security. The number of distribution structures increases by one for every distribution event. Distribution event fields include: *Distribution Code*, *Dividend Cash Amount, Factor to Adjust Price*, *Factor to Adjust Shares Outstanding*, *Distribution Declaration Date*, *Ex-Distribution Date*, *Record Date*, and *Payment Date.* The Number of Distribution Structures is greater than or equal to zero. Number of Distribution Structures can be zero if the security has no distribution history.

Primary Concept	Header Identification and Date Range Variables Distribution Event Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	NUMDIS
Database Format(s)	CA97, SFA	Data Type(s)	STK

Number of Index Types

Number of Index Types is the number of indexes or portfolio time series available for *Permanent Index Identification Numbers* in an index set. In a series set, the Number of Index Types is always one. In a group set, the Number of Index Types is always 17. Not all *Permanent Index Identification Numbers* have data for all available time series. If there is no data for one of the available time series, *Begin Index of Valid Data* and *End Index of Valid Data* of that time series are set to 0.

Primary Concept	Time Series Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Set Structure	ind	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	indtypes	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Number of List Types

Number of List Types is the number of lists available for *Permanent Index Identification Numbers* in an index set. It is set to one in index group and index series sets. If there is no data for a list, Number of Available Array Elements for the listis set to 0.

Primary Concept	Index List Arary History		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Set Structures	ind	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	listtypes	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Number of Name Structures

Number of Name Structures is the number of name history observations available in the Name History Array for a security. A name history observation is added any time there is a change to any of the name history array variables, which include: *Name Date*, *CUSIP*, *Ticker Symbol*, *Company Name*, *Share Class*, *Share Code*, *Exchange Code*, and *Standard Industrial Classification (SIC) Code*. Number of Name Structures is greater than or equal to one.

Primary Concept	Header Identification and Date Range Variables Name History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	NUMNAM
Database Format(s)	CA97, SFA	Data Type(s)	STK

Number of Nasdaq Information Structures

Number of Nasdaq Information Structures is the number of Nasdaq information structures in the Nasdaq information array for the current security. A Nasdaq information observation is added any time there is a change to any of the Nasdaq information array variables, which include: *Nasdaq Traits Date*, *Nasdaq Traits Code*, *Nasdaq National Market Indicator*, *Market Maker Count*, and *Index Level on Nasdaq Composite*. Number of Nasdaq Information Structures is zero for all Nasdaq securities that stopped trading prior to November 1, 1982 and for all NYSE/AMEX securities.

Primary Concept	Header Identification and Date Range Variables Nasdaq Information Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	NUMNDI
Database Format(s)	CA97, SFA	Data Type(s)	STK

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	Base CRSPAccess97 Data Structure		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_CAL	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	ndays	FORTRAN Variable	NDAYS
Database Format(s)	CA97	Data Type(s)	STK, IND

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 nine in daily databases and eight in monthly databases. If there is no data for a portfolio time series *Begin Index of Valid Data* and *End Index of Valid Data* are both set to 0.

Primary Concept	Portfolio Statistics and Assignment Arrays Shares Outstanding Observations Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Set Structure	stk	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	porttypes	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK*

Number of Rebalancing Types

Number of Rebalancing Types is the number of portfolio rebalancing arrays available for *Permanent Index Identification Numbers* in an index set. In a series set, Number of Rebalancing Types always = 1. In a group set Number of Rebalancing Types is always ten. Not all *Permanent Index Identification Numbers* have rebalancing data for all available portfolios. If there is no rebalancing data for one of the available rebalancing series, *Number of Array Elements Sereies* for that array is set to zero.

Primary Concept	Rebalancing History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Set Structure	ind	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	rebaltypes	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Number of Shares Structures

Number of Shares Structures is the number of shares outstanding observations in the share outstanding observation arrays for the current security. A new observation is recorded either from a company report or is imputed from a distribution event. Shares observation fields include: *Shares Outstanding*, *Shares Observation Date*, and *Shares Outstanding Observation Flag*. Number of Shares Structures is greater than or equal to zero. Number of Shares Structures can be zero if no share information is available.

Primary Concept	Header Identification and Data Range Variables		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hr, /hrl
C Object	n/a	FORTRAN Common Block	/HEADER/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Variable	NUMSHR
Database Format(s)	CA97, SFA	Data Type(s)	STK

Number of Trades, Nasdaq

Daily: Number of Trades, Nasdaq 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 paper trades are included in the count. If the number of trades is unavailable, the field is set to –99.

Number of trades is only available 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. Data is 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	Supplemental Nasdaq Time Series		
ts_print Daily Usage	`numtrd/0`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/pn`
C Object	`numtrd_ts`	FORTRAN Common Block	`NMSDAT`
C Array	`numtrd[]`	FORTRAN Array	`NMSTRD()`
C Element	`numtrd`	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Object Array

Object Array identifies the generic array used to store time series, event, or header data in CRSPAccess97 object data structures.

Primary Concept	Base CRSPAccess97 Data Structure		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`

C Object Type	`CRSP_TIMESERIES`
	`CRSP_ARRAY`
	`CRSP_ROW`
	`CRSP_CAL`
		FORTRAN Common Block	`n/a`

C Array	`n/a`	FORTRAN Array	`n/a`
C Element	`arr`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`STK, IND`

Object Type Code

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

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

Primary Concept	Base CRSPAccess97 Data Structures		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object Type	CRSP_TIMESERIES CRSP_ARRAY CRSP_ROW CSRP_CAL	FORTRAN Common Block	n/a
C Array	n/a	FORTRAN Array	n/a
C Element	objtype	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK, IND

Optional Time Series 1

Optional Time Series 1 is a time series selected with conversion utilties or programming loading options. In the SFA Guide, see "Chapter 3. CRSPAccess97 Format to SFA Format Conversion" on page 43 for CRSPAccess97 to SFA stock file conversion utilities and see the WANTED description in the CRSPAccess97 Database Format - Programmers Guide page 23 for load options.

Primary Concept	Optional Time Series Data Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/ADATA/
C Array	n/a	FORTRAN Array	SXRET()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	STK (STK* if excess return)

Optional Time Series 2

Optional Time Series 2 is a time series selected with conversion utilties or programming loading options. In the SFA Guide, see "Chapter 3. CRSPAccess97 Format to SFA Format Conversion" on page 43 for CRSPAccess97 to SFA stock file conversion utilities and see the WANTED description in the CRSPAccess97 Database Format - Programmers Guide page 23 for load options.

Primary Concept	Optional Data Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/ADATA/
C Array	n/a	FORTRAN Array	BXRET()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	STK*

Partition Subset Screening Structure

Partition Subset Screening Structure, like the *Index 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*. Partition Subset Screening Structure screens are used to restrict the securities used in defining partition breakpoints of an index.

Primary Concept	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Structure	induniv	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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 0 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	Distribution Event Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/di
C Object	dists_arr	FORTRAN Common Block	/INFO/
C Array	dists[]	FORTRAN Array	DISTS(,)
C Element	paydt	FORTRAN Element	PAYDT
Database Format(s)	CA97, SFA	Data Type(s)	STK

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	Header Identification and Summary Data Header Identification and Date Range Variables		
ts_print Daily Usage	permco/0		
ts_print Monthly Usage	mpermco/0	*stk_print* Option(s)	/hh, /hr, /hrl
C Object	header_arr	FORTRAN Common Block	/HEADER/
C Array	header	FORTRAN Array	n/a
C Element	permco	FORTRAN variable	PERMCO
Database Format(s)	CA97, SFA	Data Type(s)	STK

Permanent Index Group Identification Number

Permanent Index Group Identification Number 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	Index Header		
ts_print Daily Usage	permco		
ts_print Monthly Usage	mpermco	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	indco	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Permanent Index Identification Number

Permanent Index Identification Number is the unique permanent identifier assigned by CRSP to every supported index. All Permanent Index Identification Numbers are seven-digit integers. There is no inherent meaning in the numbers. The indices sets in a CRSPAccess97 database are sorted by this field. See "3.2 CRSP Index Series and Groups" on page 51 for a full list of CRSP Indices.

Primary Concept	Index Header		
ts_print Daily Usage	permno		
ts_print Monthly Usage	mpermno	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	indno	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Permanent Index Identification Number of Associated Index

Permanent Index Identification Number of Associated Index is the *Permanent Index Identification Number* of an associated index used to supply rebalancing breakpoint information used for assignments or buy/sell rules to this index. It is set to 0 if external porfolio data is used.

Primary Concept	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	asperm	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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 is sorted and indexed by this field**. PERMNO is currently a five-digit integer for all common securities in the CRSP files. The range -999989 to -1000 and 1000 to 999989 is reserved for CRSP PERMNO assignments.

Primary Concept	Header Identification and Summary Data Header Identification and Data Range Variables		
ts_print Daily Usage	permno		
ts_print Monthly Usage	mpermno	*stk_print* Option(s)	/hh, /hr, /hrl
C Object	header_row	FORTRAN Common Block	HEADER
C Array	header	FORTRAN Array	n/a
C Element	permno	FORTRAN Variable	PERMNO
Database Format(s)	CA97, SFA	Data Type(s)	STK

Permanent Number of Securities in Index List

Permanent Number of Securities in Index List is the CRSP PERMNO of a member security of an index specified with an Index List Array.

Primary Concept	Index List History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	list_arr[]	FORTRAN Common Block	n/a
C Array	list[][]	FORTRAN Array	n/a
C Element	permno	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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 45. 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	Portfolio Assignment Array		
ts_print Daily Usage	port/0		
ts_print Monthly Usage	mport/0	*stk_print* Option(s)	/dy
C Object	port_ts[]	FORTRAN Common Block	n/a
C Array	port[][]	FORTRAN Array	n/a
C Element	port	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK‡

Portfolio Assignment for Betas

daily: Portfolio Assignment for Betas is the portfolio ranking for a security when the third portfolio type loaded is either NYSE/AMEX or Nasdaq Beta Deciles. See "Portfolio Assignment for Third Portfolio" on page 120, see "Portfolio Assignment Array" on page 42 and see "Chapter 3: CRSP Index Methodologies" on page 45 for additional information.

Primary Concept	Portfolio Assignment Array		
ts_print Daily Usage	`port/6` or `port/8`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/dy6` or `/dy8`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Assignment Number	FORTRAN Array	`PRTNUM(BETA,)`
C Equivalent Usage	`port[i][].port` where `i=5` or `7`	FORTRAN Element	`BETA`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK*`

Portfolio Assignment for Capitalizations

Portfolio Assignment for Capitalizations is the portfolio ranking for a security when the first portfolio type loaded is the NYSE/AMEX/Nasdaq Capitalization Decile Indices. See "Portfolio Assignment for First Portfolio" on page 118, see "Portfolio Assignment Array" on page 42 and see "Chapter 3: CRSP Index Methodologies" on page 45 for additional information.

Primary Concept	Portfolio Assignment Array		
ts_print Daily Usage	`port/1`		
ts_print Monthly Usage	`mport/1`	*stk_print* Option(s)	`/dy1`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Assignment Number	FORTRAN Array	`PRTNUM(CAP,)`
C Equivalent Usage	`port[0][].port`	FORTRAN Element	`CAP`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK*`

Portfolio Assignment for First Portfolio

Portfolio Assignment for First Portfolio is the annual portfolio ranking for a security in the first portfolio type loaded. The actual data loaded is controlled by the user. See "3.3 Portfolio Types Defined by CRSP" on page 54 for a list of annual portfolio types. See "Chapter 3: CRSP Index Methodologies" on page 45, for methodologies used to make assignments to the defined CRSP portfolio types.

Primary Concept	Portfolio Assignment Array		
ts_print Daily Usage	`port/#`		
ts_print Monthly Usage	`mport/#`	*stk_print* Option(s)	`/dy#`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Assignment Number	FORTRAN Array	`PRTNUM(1,)`
C Equivalent Usage	`port[i][].port` where `i+1` is the portfolio type	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK*`

Portfolio Assignment for Nasdaq Capitalizations

Portfolio Assignment for Nasdaq Capitalizations is the portfolio ranking for a security when the third portfolio type loaded is Nasdaq Capitalization Deciles. See "Portfolio Assignment for Third Portfolio" on page 120, see "Portfolio Assignment Array" on page 42 and see "Chapter 3: CRSP Index Methodologies" on page 45 for additional information.

Primary Concept	Portfolio Assignment Array		
ts_print Daily Usage	`port/3`		
ts_print Monthly Usage	`mport/3`	*stk_print* Option(s)	`/dy3`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Assignment Number	FORTRAN Array	`PRTNUM(CAPQ,)`
C Equivalent Usage	`port[2][].port`	FORTRAN Element	`CAPQ`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK‡`

Portfolio Assignment for NYSE/AMEX Capitalizations

Portfolio Assignment for NYSE/AMEX Capitalizations is the portfolio ranking for a security when the second portfolio type loaded is NYSE/AMEX Capitalization Deciles. See "Portfolio Assignment for Second Portfolio" on page 119, see "Portfolio Assignment Array" on page 42 and see "Chapter 3: CRSP Index Methodologies" on page 45 for additional information.

Primary Concept	Portfolio Assignment Array		
ts_print Daily Usage	`port/2`		
ts_print Monthly Usage	`mport/2`	*stk_print* Option(s)	`/dy2`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Assignment Number	FORTRAN Array	`PRTNUM(CAPN,)`
C Equivalent Usage	`port[1][].port`	FORTRAN Element	`CAPN`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK‡`

Portfolio Assignment for Second Portfolio

Portfolio Assignment for Second Portfolio is the annual portfolio ranking for a security in the second portfolio type loaded. The actual data loaded is controlled by the user. See "3.3 Portfolio Types Defined by CRSP" on page 54 for a list of annual portfolio types. See "Chapter 3: CRSP Index Methodologies" on page 45, for methodologies used to make assignments to the defined CRSP portfolio types.

Primary Concept	Portfolio Assignment Array		
ts_print Daily Usage	`port/#`		
ts_print Monthly Usage	`mport/#`	*stk_print* Option(s)	`/dy#`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Assignment Number	FORTRAN Array	`PRTNUM(2,)`
C Equivalent Usage	`port[i][].port` where `i+1` is the portfolio type	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK‡`

Portfolio Assignment for Standard Deviations

daily: Portfolio Assignment for Standard Deviation Excess Returns is the portfolio ranking for a security when the second portfolio type loaded is standard deviation portfolio data. See "Portfolio Assignment for Second Portfolio" on page 119, see "Portfolio Assignment Array" on page 42 and see "Chapter 3: CRSP Index Methodologies" on page 45 for additional information.

Primary Concept	Portfolio Assignment Array		
ts_print Daily Usage	`port/7` or `port/9`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/dy7` or `/dy9`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Assignment Number	FORTRAN Array	`PRTNUM(SDEV,)`
C Equivalent Usage	`port[i][].port` where `i` is 6 or 8	FORTRAN Element	`SDEV`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK*`

Portfolio Assignment for Third Portfolio

Portfolio Assignment for Third Portfolio is the annual portfolio ranking for a security in the third portfolio type loaded. The actual data loaded is controlled by the user. See "3.3 Portfolio Types Defined by CRSP" on page 54 for a list of annual portfolio types. See "Chapter 3: CRSP Index Methodologies" on page 45, for methodologies used to make assignments to the defined CRSP portfolio types.

Primary Concept	Portfolio Assignment Array		
ts_print Daily Usage	`port/#`		
ts_print Monthly Usage	`mport/#`	*stk_print* Option(s)	`/dy#`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Assignment Number	FORTRAN Array	`PRTNUM(3,)`
C Equivalent Usage	`port[i][].port` where `i+1` is the portfolio type	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK‡`

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	Index Header		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`indhdr_row`	FORTRAN Common Block	`n/a`
C Array	`indhdr`	FORTRAN Array	`n/a`
C Element	`rules`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`IND`

Portfolio Company Count

Portfolio Company Count is the number of companies included in a CRSP Cap-Based portfolio at the beginning of quarter *Year and Month of Quarter.* There are 10 portfolios based on NYSE deciles included, with the largest companies in portfolio 1 and the smallest in portfolio 10.

Primary Concept	Cap-Based Reports Rebalancing History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/REBAL/
C Array	n/a	FORTRAN Array	PRTCCT(,)
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Portfolio Issue Count

Portfolio Issue Count contains the number of issues included in the selected portfolio on the selected *Calendar Trading Date* for a CRSP Cap-Based Portfolio.

Primary Concept	Cap-Based Reports Monthly History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAPBAS/
C Array	n/a	FORTRAN Array	PRTCNT(,)
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Portfolio Largest Company Name

Portfolio Largest Company Name is the name of the largest company included in a CRSP Cap-Based portfolio at the beginning of quarter *Year and Month of Quarter*.

Primary Concept	Cap-Based Reports Rebalancing History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/REBAL/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Element	MAXCNM(,)
Database Format(s)	SFA	Data Type(s)	IND

Portfolio Number if Subset Series

Portfolio Number if Subset Series is the portfolio number within an index group this index series belongs to. 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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	portnum	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Portfolio Number in Associated Index

Portfolio Number in Associated Index is the portfolio number within an associated index group defined in *Permanent Index Identification Number of Associated Index*. The associated index breakpoint information for that portfolio is used for this index. It is set to 0 if no outside rebalancing information is used to build this index.

Primary Concept	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	asport	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Portfolio Number of Decile

Portfolio Number of Decile is the decile portfolio number of a CRSP Cap-Based portfolio at the beginning of quarter *Year And Month of Quarter.* There are 10 portfolios based on NYSE deciles included, with the largest companies in portfolio 1 and the smallest in portfolio 10.

Primary Concept	Index Rebalancing History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/REBAL/
C Array	n/a	FORTRAN Array	n/a
C Element	n/a	FORTRAN Element	PRTNO(,)
Database Format(s)	CA97, SFA	Data Type(s)	IND

Portfolio Sequence Number

Portfolio Sequence Number contains a short character string describing the selected Cap-Based portfolio. The first ten based on NYSE deciles are numbered 1 - 10, with the largest companies in portfolio 1 and the smallest in portfolio 10. Portfolios 11 - 17 are composites of the first ten. The sequence of the portfolios is listed in the following table.

Portfolio Sequence Number	Name	Sequence
1	CRSP 1	1
2	CRSP 2	2
3	CRSP 3	3
4	CRSP 4	4
5	CRSP 5	5
6	CRSP 6	6
7	CRSP 7	7
8	CRSP 8	8
9	CRSP 9	9
10	CRSP 10	10
1-2	CRSP 1-2	11
3_5	CRSP 3-5	12
6_8	CRSP 6-8	13
9_10	CRSP 9-10	14
1_5	CRSP 1-5	15
6_10	CRSP 6-10	16
Mrkt	CRSP Market	17

Primary Concept	Cap-Based Reports Monthly History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAPBAS/
C Array	n/a	FORTRAN Array	PRTNAM(,)
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	SFA	Data Type(s)	IND

Portfolio Smallest Company Name

Portfolio Smallest Company Name is the name of the smallest company included in a CRSP Cap-Based portfolio at the beginning of quarter *Year And Month of Quarter.*

Primary Concept	Cap-Based Reports Rebalancing History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/REBAL/
C Array	n/a	FORTRAN Array	MINCNM(,)
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

minus one

Portfolio Statistic for Betas

daily: Portfolio Statistic for Betas is the portfolio ranking for a security when the third portfolio type loaded is either NYSE/AMEX or Nasdaq Beta Deciles. See "Portfolios Statistic for Third Portfolio Type" on page 126, see "Portfolio Statistic Array" on page 43 and see "Chapter 3: CRSP Index Methodologies" on page 45 for additional information.

Primary Concept	Portfolio Statistic Array		
ts_print Daily Usage	`stat/6` or `stat/8`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/dy6` or `/dy8`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Statistic Value	FORTRAN Array	`YRVAL(BETA,)`
C Equivalent Usage	`port[i][].stat` where `i=5` or `7`	FORTRAN Element	`BETA`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK‡`

Portfolio Statistic for Capitalizations

Portfolio Statistic for Capitalizations is the portfolio ranking for a security when the first portfolio type loaded is the NYSE/AMEX/Nasdaq Capitalization Decile Indices. See "Portfolio Statistic for First Portfolio Type" on page 124, see "Portfolio Statistic Array" on page 43 and see "Chapter 3: CRSP Index Methodologies" on page 45 for additional information.

Primary Concept	Portfolio Statistic Array		
ts_print Daily Usage	`stat/1`		
ts_print Monthly Usage	`mstat/1`	*stk_print* Option(s)	`/dy1`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Statistic Value	FORTRAN Array	`YRVAL(CAP,)`
C Equivalent Usage	`port[0][].stat`	FORTRAN Element	`CAP`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK*`

Portfolio Statistic for First Portfolio Type

Portfolio Statistic for First Portfolio is the annual portfolio ranking for a security in the first portfolio type loaded. The actual data loaded is controlled by the user. See "3.3 Portfolio Types Defined by CRSP" on page 54 for a list of annual portfolio types. See "Chapter 3: CRSP Index Methodologies" on page 45, for methodologies used to make assignments to the defined CRSP portfolio types.

Primary Concept	Portfolio Statistic Array		
ts_print Daily Usage	`stat/#`		
ts_print Monthly Usage	`mstat/#`	*stk_print* Option(s)	`/dy#`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Statistic Value	FORTRAN Array	`YRVAL(1,)`
C Equivalent Usage	`port[i][].stat` where `i+1` is the portfolio type	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK*`

Portfolio Statistic for Nasdaq Capitalizations

Portfolio Statistic for Nasdaq Capitalizations is the portfolio ranking for a security when the third portfolio type loaded is Nasdaq Capitalization Deciles. See "Portfolios Statistic for Third Portfolio Type" on page 126, see "Portfolio Statistic Array" on page 43 and see "Chapter 3: CRSP Index Methodologies" on page 45 for additional information.

Primary Concept	Portfolio Statistic Array		
ts_print Daily Usage	`stat/3`		
ts_print Monthly Usage	`mstat/3`	*stk_print* Option(s)	`/dy3`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Statistic Value	FORTRAN Array	`YRVAL(CAPQ)`
C Equivalent Usage	`port[2][].stat`	FORTRAN Element	`CAPQ`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK‡`

Portfolio Statistic for NYSE/AMEX Capitalizations

Portfolio Statistic for NYSE/AMEX Capitalizations is the portfolio ranking for a security when the second portfolio type loaded is NYSE/AMEX Capitalization Deciles. See "Portfolio Statistic for Second Portfolio Type" on page 125, see "Portfolio Statistic Array" on page 43 and see "Chapter 3: CRSP Index Methodologies" on page 45 for additional information.

Primary Concept	Portfolio Statistic Array		
ts_print Daily Usage	`stat/2`		
ts_print Monthly Usage	`mstat/2`	*stk_print* Option(s)	`/dy2`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Statistic Value	FORTRAN Array	`YRVAL(CAPN,)`
C Equivalent Usage	`port[1][].stat`	FORTRAN Element	`CAPN`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK‡`

Portfolio Statistic for Second Portfolio Type

Portfolio Statistic for Second Portfolio is the annual portfolio ranking for a security in the second portfolio type loaded. The actual data loaded is controlled by the user. See "3.3 Portfolio Types Defined by CRSP" on page 54 for a list of annual portfolio types. See "Chapter 3: CRSP Index Methodologies" on page 45, for methodologies used to make assignments to the defined CRSP portfolio types.

Primary Concept	Portfolio Statistic Array		
ts_print Daily Usage	`stat/#`		
ts_print Monthly Usage	`mstat/#`	*stk_print* Option(s)	`/dy#`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Statistic Value	FORTRAN Array	`YRVAL(2,)`
C Eqivalent Usage	`port[i][].stat` where `i+1` is the portfolio type	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK‡`

Portfolio Statistic for Standard Deviations

daily: Portfolio Statistic for Standard Deviation Excess Returns is the portfolio ranking for a security when the second portfolio type loaded is standard deviation portfolio data. See "Portfolio Statistic for Second Portfolio Type" on page 125, see "Portfolio Statistic Array" on page 43 and see "Chapter 3: CRSP Index Methodologies" on page 45 for additional information.

Primary Concept	Portfolio Statistic Array		
ts_print Daily Usage	`stat/7` or `stat/9`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/dy7` or `/dy9`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Variable Equivalent	Portfolio Statistic Value	FORTRAN Array	`YRVAL(SDEV)`
C Equivalent Usage	`port[i][].stat` where `i` is 6 or 8	FORTRAN Element	`SDEV`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK‡`

Portfolios Statistic for Third Portfolio Type

Portfolio Statistic for Third Portfolio is the annual portfolio ranking for a security in the third portfolio type loaded. The actual data loaded is controlled by the user. See "3.3 Portfolio Types Defined by CRSP" on page 54 for a list of annual portfolio types. See "Chapter 3: CRSP Index Methodologies" on page 45, for methodologies used to make assignments to the defined CRSP portfolio types.

Primary Concept	Portfolio Statistic Array		
ts_print Daily Usage	`stat/#`		
ts_print Monthly Usage	`mstat/#`	*stk_print* Option(s)	`/dy#`
C Object	`n/a`	FORTRAN Common Block	`/ADATA/`
C Varaible Equivalent	Portfolio Statistic Value	FORTRAN Array	`YRVAL(3,)`
C Equivalent Usage	`port[i][].stat` where `i+1` is the portfolio type	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK‡`

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	Portfolio Statistics and Assignment Time Series		
ts_print Daily Usage	`port/#`		
ts_print Monthly Usage	`mport/#`	*stk_print* Option(s)	`/dy`
C Object	`port_ts[]`	FORTRAN Common Block	`n/a`
C Array	`port[][]`	FORTRAN Variable Equivalent	all Portfolio Statistic for ... variables
C Element	`stat`	FORTRAN Equivalent Usage	`YRVAL(,)`
Database Format(s)	`CA97`	Data Type(s)	`STK*`

Portfolio Weight

Portfolio Weight contains the total capitalization in 1000's of dollars included on a given date for a CRSP Cap-Based portfolio. *Portfolio Sequence Number* describes the portfolio. Weight is as of the end of the previous month.

Primary Concept	Cap-Based Reports Monthly History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAPBAS/
C Variable Equivalent	Portfolio Used Weight Arrays	FORTRAN Array	PRTWGT(,)
C Equivalent Usage	usdval[][]	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

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* .

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

Primary Concept	Auxiliary Time Series		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	altprc_ts	FORTRAN Common Block	n/a
C Array	altprc[]	FORTRAN Array	n/a
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK

Price Alternate Date

Monthly: Price Alternate Date contains the date of the monthy *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	Supplemental Nasdaq Data		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	numtrd_ts	FORTRAN Common Block	/NMSDAT/
C Array	numtrd[]	FORTRAN Array	NMSTRD()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK

Price or Bid/Ask Average

Price or Bid/Ask Average is the closing price or the bid/ask average for a trading period. To differentiate them, prices have positive values and bid/ask averages have negative values. If the closing price is not available for any given period, the number in the price field is a bid/ask average, not an actual closing price. Please note that in this field the negative sign is a symbol and that the value of the bid/ask average is not negative. If neither price nor bid/ask average is available on a date, Price or Bid/Ask Average is set to zero.

If the security of a company is included in the Composite Pricing network, the closing price while listed on NYSE or AMEX on a trading date is the last trading price for that day on the exchange that the security last traded. Similarly, highs, lows, and volumes include trades on all 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 comes directly from the NASD with the close of the day at 4:00 p.m. Eastern Time. Automated trades after hours on Nasdaq are counted on the next trading date, although the volumes are applied to the current date.

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 Small Cap 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	Price, Volume, and Return Arrays		
ts_print Daily Usage	`prc/0`		
ts_print Monthly Usage	`mprc/0`	*stk_print* Option(s)	`/pp, /dd, /dr, /dx, /ds`
C Object	`prc_ts`	FORTRAN Common Block	`/DDATA/`
C Array	`prc[]`	FORTRAN Array	`PRC()`
C Element	`n/a`	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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 where 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	Event Arrays/Distribution History Array		
ts_print Daily Usage			
ts_print Monthly Usage		*stk_print* Option(s)	`/di`
C Object	`dists_arr`	FORTRAN Common Block	`/INFO/`
C Array	`dists[]`	FORTRAN Array	`DISTS(,)`
C Element	`rcrddt`	FORTRAN Element	`RCRDDT`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Related Assignment Information

Related Assignment Information is a group of fields defining the time periods and associated indexes used to form portfolios. It primarily defines the rebalancing periods when the portfolio is reformed based on new information. Related Assignment Information contains fields *Basic Assignment Type Code*, *Acquiring `PERMNO`*, *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	Index Header		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`indhdr_row`	FORTRAN Common Block	`n/a`
C Array	`indhdr`	FORTRAN Array	`n/a`
C Structure	`assign`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`IND`

Restriction Begin Date (Partition or Index)

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

Primary Concept	Index Header		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`indhdr_row`	FORTRAN Common Block	`n/a`
C Array	`indhdr`	FORTRAN Array	`n/a`
C Element	`begdt`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`IND`

Restriction End Date (Partition or Index)

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

Primary Concept	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	enddt	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Return (Excluding Dividends) on Equal-Weighted Index

Return (Excluding Dividends) on Equal-Weighted Index contains returns, excluding all dividends, on an equally-weighted market portfolio (including ADRs).

Primary Concept	Calendar/Indices, CRSP Stock File Indices		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAL/
C Array	n/a	FORTRAN Array	EWRETX()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	STK, IND

Return (Excluding Dividends) on Value-Weighted Index

Return (Excluding Dividends) on Value-Weighted Index contains, returns, excluding all dividends, on a value-weighted market portfolio (excluding ADRs).

Primary Concept	Calendar/Indices, Decile Returns, and Index Levels Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAL/
C Array	n/a	FORTRAN Array	VWRETX()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	STK*, IND

Return (Including all Distributions) on Equal-Weighted Index

Return (Including all Distributions) on Equal-Weighted Index contains returns, including all distributions, on an equally-weighted market portfolio (including ADRs).

Primary Concept	Calendar/Indices, CRSP Stock File Indices		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAL/
C Array	n/a	FORTRAN Array	EWRETD()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	STK*, IND

Return (Including all Distributions) on Value-Weighted Index

Return (Including all Distributions) on Value-Weighted Index contains the returns, including all distributions, on a value-weighted market portfolio (excluding (ADRs)).

Primary Concept	Calendar/Indices, Decile Returns, and Index Levels Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAL/
C Array	n/a	FORTRAN Array	VWRETD()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	STK*, IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	delretflag	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Return on 1-Year Bonds

The Return on 1-Year Bonds is the return during the calendar period of the selected 1-Year Bond. Valid returns require a price the previous period and current period and are calculated by dividing the current price plus interest payments by the previous price. This data is available beginning January 31, 1941. Values prior to this date are set to –99.

Primary Concept	CTI Series		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	mret/0 and mindret/1000706	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	B1RET()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Return on 10-Year Bonds

The Return on 10-Year Bonds is the return during the calendar period of the selected 10-Year Bond. Valid returns require a price the previous period and current period and are calculated by dividing the current price plus interest payments by the previous price. This data is available beginning May 31, 1941. Values prior to this date are set to –99.

Primary Concept	CTI Series		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	mret/0 and mindret/1000702	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	B10RET()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Return on 2-Year Bonds

The Return on 2-Year Bonds is the return during the calendar period of the selected 2-Year Bond. Valid returns require a price the previous period and current period and are calculated by dividing the current price plus interest payments by the previous price. This data is available beginning January 31, 1941. Values prior to this date are set to –99.

Primary Concept	CTI Series		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	mret/0 and mindret/1000705	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	B2RET()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Return on 20-Year Bonds

The Return on 20-Year Bonds is the return during the calendar period of the selected 20-Year Bond. Valid returns require a price the previous period and current period and are calculated by dividing the current price plus interest payments by the previous price. This data is available beginning January 31, 1942. Values prior to this date are set to –99.

Primary Concept	CTI Series		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	mret/0 and mindret/1000701	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	B20RET()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Return on 30-Day Bills

The Return on 30-Day Bills is the return during the calendar period of the selected 30-Day Bills. Valid returns require a price the previous period and current period and are calculated by dividing the current price plus interest payments by the previous price. The scarcity of available issues prior to 1937 resulted in the use of some very long nominal one month issues. The range of maturities after 1937 is within a few days of the 30-day target, and users may wish to restrict their usage of Return on 30-Day Bills and Index Level Associated with Return on 30-Day Bills to this period. Values prior to this date are set to –99.

Primary Concept	CTI Series		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`mret/0 and mindret/1000701`	*stk_print* Option(s)	`n/a`
C Object	`n/a`	FORTRAN Common Block	`/CTIS/`
C Array	`n/a`	FORTRAN Array	`B30RET()`
C Element	`n/a`	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`IND`

Return on 30-Year Bonds

The Return on 30-Year Bonds is the return during the calendar period of the selected 30-Year Bond. Valid returns require a price the previous period and current period and are calculated by dividing the current price plus interest payments by the previous price. This data is available beginning November 29, 1941. Values prior to this date are set to –99.

Primary Concept	CTI Series		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`mret/0 and mindret/1000700`	*stk_print* Option(s)	`n/a`
C Object	`n/a`	FORTRAN Common Block	`/CTIS/`
C Array	`n/a`	FORTRAN Array	`B30RET()`
C Element	`n/a`	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`IND`

Return on 5-Year Bonds

The Return on 5-Year Bonds is the return during the calendar period of the selected 5-Year Bond. Valid returns require a price the previous period and current period and are calculated by dividing the current price plus interest payments by the previous price. This data is available beginning April 30, 1941. Values prior to this date are set to –99.

Primary Concept	CTI Series		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`mret/0 and mindret/1000704`	*stk_print* Option(s)	`n/a`
C Object	`n/a`	FORTRAN Common Block	`/CTIS/`
C Array	`n/a`	FORTRAN Array	`B5RET()`
C Element	`n/a`	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`IND`

Return on 7-Year Bonds

The Return on 7-Year Bonds is the return during the calendar period of the selected 7-Year Bond. Valid returns require a price the previous period and current period and are calculated by dividing the current price plus interest payments by the previous price. This data is available beginning April 30, 1941. Values prior to this date are set to –99.

Primary Concept	CTI Series		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	mret/0 and mindret/1000703	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	B7RET()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Return on 90-Day Bills

The Return on 90-Day Bills is the return during the calendar period of the selected 90-Day Bills. Valid returns require a price the previous period and current period and are calculated by dividing the current price plus interest payments by the previous price. The scarcity of available issues prior to 1942 resulted in the use of some very long nominal 90-day issues. The range of maturities after 1942 is within a few days of the 90-day target, and users may wish to restrict their usage of Return on 90-Day Bills and Index Level Associated with Return on 90-Day Bills to this period. Values prior to this date are set to –99.

Primary Concept	CTI Series		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	mret/0 and mindret/1000707	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CTIS/
C Array	n/a	FORTRAN Array	B90RET()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Return on Decile

Return on Decile is the return on the selected market segment decile portfolio on the requested *Calendar Trading Date*.

Primary Concept	Calendar/Indices, Decile Returns, Index Levels Arrays Stock File Capitalization Deciles, Stock File Risk-Based Deciles		
ts_print Daily Usage	ret/0 for series indno		
ts_print Monthly Usage	mret/0 for series indno	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/DECILE/
C Array	n/a	FORTRAN Array	DECRET(,)
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Return on Income Portfolio

Return on Income Portfolio is the income return on the selected portfolio on the selected *Calendar Trading Date* for a CRSP Cap-Based Portfolio.

Primary Concept	Cap-Based Reports Monthly History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	mreti/0 for cap-based indno	*stk_print* Option(s)	
C Object	n/a	FORTRAN Common Block	/CAPBAS/
C Array	n/a	FORTRAN Array	INCRET(,)
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Return on Nasdaq Composite Index

Return on Nasdaq Composite Index is the return on the Nasdaq Composite Index over the calendar period, defined as:

$$\frac{\text{Index of Nasdaq Composite}}{\text{Index of Nasdaq Composite at the end of the previous period}} \text{ minus one}$$

This represents the Returns without Dividends of the Permanent Index Identification Number 1000503.

Primary Concept	CRSP Stock File Indices, Market Indices		
ts_print Daily Usage	ret/0 for indno 1000503		
ts_print Monthly Usage	maret for indno 1000503	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAL/
C Array	n/a	FORTRAN Array	NCRTRN()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	STK, IND

Return on Portfolio

monthly: Return on Portfolio is the total monthly holding period return of a CRSP Cap-Based portfolio.

Primary Concept	Cap-Based Reports Monthly History Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a, see mret/ or mindtret/indno	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAPBAS/
C Array	n/a	FORTRAN Array	TOTRET()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

Return without Dividends

Return without Dividends contains returns without dividends, or capital appreciation of a security. Ordinary dividends and certain other regularly taxable dividends are excluded from the returns calculation. See *Holding Period Total Returns* 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	Auxiliary Data Arrays Optional Data Arrays		
ts_print Daily Usage	`retx/0`		
ts_print Monthly Usage	`mretx/0`	*stk_print* Option(s)	`/px, /dr`
C Object	`retx_ts`	FORTRAN Common Block	`/ADATA/`
C Array	`retx[]`	FORTRAN Array	`RETX()`
C Element	`n/a`	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

S&P 500 Composite Index Level

The S&P 500 Composite Index is the level of the *S&P 500 Composite Index Return* (prior to March 1957, this was the S&P 90-stock index) at the end of the trading day or month. This data is collected from publicly available sources such as the Dow Jones News Service, *The Wall Street Journal* or *Standard & Poor's Statistical Service*. The S&P 500 Composite Index does not include dividends. The index indicates the change in price of the component securities.

Primary Concept	Calendar/Indices, Published Standard & Poor's 500 and Nasdaq Composite Data		
ts_print Daily Usage	`aind for indno 1000502`		
ts_print Monthly Usage	`maind for indno 1000502`	*stk_print* Option(s)	`n/a`
C Object	`n/a`	FORTRAN Common Block	`/CAL/`
C Array	`n/a`	FORTRAN Array	`SPINDX()`
C Element	`n/a`	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK, IND`

S&P 500 Composite Index Return

The S&P 500 Composite Index Return is the return on the Standard & Poor's Composite Index defined as:

$$\frac{\text{Index of S\&P 500 Composite}}{\text{Index of S\&P 500 Composite at the end of the previous period}} \text{ minus one}$$

This represents the Capital Appreciation of the Permanent Index Identification Number 1000503.

Primary Concept	Calendar/Indices, Published Standard & Poor's 500 and Nasdaq Composite Indices Data		
ts_print Daily Usage	`aret for indno 1000502`		
ts_print Monthly Usage	`market for indno 1000502`	*stk_print* Option(s)	`n/a`
C Object	`n/a`	FORTRAN Common Block	`/CAL/`
C Array	`n/a`	FORTRAN Array	`SPRTRN()`
C Element	`n/a`	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK, IND`

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	Name History Array		
ts_print Daily Usage	`shrcls/0`		
ts_print Monthly Usage	`mshrcls/0`	*stk_print* Option(s)	`/n`
C Object	`names_arr`	FORTRAN Common Block	`/INFO/`
C Array	`names[]`	FORTRAN Array	`CNAMES(,)`
C Element	`shrcls`	FORTRAN Element	`SHRCLS`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Share Code

Share Code is a two-digit 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
3	ADRs (American Depository Receipts)
4	SBIs (Shares of Beneficial Interest)
7	Units (Depository Units, Units of Beneficial Interest, Units of Limited Partnership Interest, Depository Receipts, etc.)

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 (Primes and Scores).
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	Name History Array		
ts_print Daily Usage	`shrcd/0`		
ts_print Monthly Usage	`mshrcd/0`	*stk_print* Option(s)	`/n`
C Object	`names_arr`	FORTRAN Common Block	`/INFO/`
C Array	`names[]`	FORTRAN Array	`NAMES(,)`
C Element	`shrcd`	FORTRAN Element	`SHRCD`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	sccode	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Share Code, Header

Share Code, Header is the last CRSP Share Code in a specific security's Name History Array. See *Share Code* for the description of the CRSP share type coding scheme.

Primary Concept	Header Identification and Summary Data		
ts_print Daily Usage	n/a		n/a
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	header_row	FORTRAN Common Block	n/a
C Array	header	FORTRAN Array	n/a
C Element	hshrcd	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	STK

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 the definition for Share Codes for details of the two-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 Cod*e, and *Valid Second Digit of Share Code*.

Primary Concept	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	shrcd	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Shares 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	Shares Outstanding Observations Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/sh, /sa`
C Object	`shares_arr`	FORTRAN Common Block	`/INFO/`
C Array	`shares[]`	FORTRAN Array	`SHARES(,)`
C Element	`shrsdt`	FORTRAN Element	`SHRSDT`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Shares 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*.

Primary Concept	Shares Outstanding Observations Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/sh, /sa`
C Object	`shares_arr`	FORTRAN Common Block	`n/a`
C Array	`shares[]`	FORTRAN Array	`n/a`
C Element	`shrsenddt`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`STK`

Shares Outstanding

Shares Outstanding is the number of publicly held shares, recorded in thousands.

Primary Concept	Shares Outstanding Observations Array		
ts_print Daily Usage	`shr/0`		
ts_print Monthly Usage	`mshr/0`	*stk_print* Option(s)	`/sh, /sa`
C Object	`shares_arr`	FORTRAN Common Block	`/INFO/`
C Array	`shares[]`	FORTRAN Array	`SHARES(,)` or `CURSHR` function
C Element	`shrout`	FORTRAN Element	`SHROUT`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

Shares Outstanding Observation Flag

Shares Outstanding Observation Flag is an integer value indicating the source of the shares outstanding observation. A value of zero indicates a share structure extracted from CRSP data sources. A value of one corresponds to a shares structure imputed from a split or other distribution.

Primary Concept	Shares Outstanding Observations Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/sh, /sa
C Object	shares_arr	FORTRAN Common Block	/INFO/
C Array	shares[]	FORTRAN Array	SHARES(,)
C Element	shrflg	FORTRAN Element	SHRFLG
Database Format(s)	CA97, SFA	Data Type(s)	STK

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 only available when *Ask or High Price* and *Bid or Low Price* are available and *Closing Price or Bid/Ask Average* is a bid/ask average. It is set to 0 if unavailable.

Primary Concept	Auxiliary Data Time Series Optional Time Series Data		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	spread_ts	FORTRAN Common Block	/ADATA/
C Array	spread[]	FORTRAN Array	PRC2()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	Ca97, SFA	Data Type(s)	STK

Standard Deviation Excess Return

Standard Deviation Excess Return denotes the excess return of a specific issue less the average return of all issues in its standard deviation portfolio on a given day. It is computed by using the returns on the price or bid/ask average. A missing Standard Deviation Excess Return due to a portfolio assignment of zero is set to -44.0.

Primary Concept	Optional Time Series Data Arrays		
ts_print Daily Usage	portxsret/3 for NYSE/AMEX portsxret/9 for Nasdaq		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/ADATA/
C Array	n/a	FORTRAN Array	SXRET()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	STK‡

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 0. The Nasdaq stock exchange reports the first three digits of Nasdaq listed companies based on company reported primary industry, and CRSP has added a fourth digit of SIC codes of NYSE and AMEX companies are reported with four digits by IDSI based on SIC groupings. Note that new industry code, the North American Industry Classification System (NAICS) was introduced in 1997, to succeed the SIC codes. This new code allows for the addition of extra digits in the code, and the field has been expanded to support these. The new codes are not yet supported by CRSP or its data providers.

Primary Concept	Name History Array		
ts_print Daily Usage	`siccd/0`		
ts_print Monthly Usage	`msiccd/0`	*stk_print* Option(s)	`/n`
C Object	`names_arr`	FORTRAN Common Block	`/INFO/`
C Array	`names[]`	FORTRAN Array	`NAMES(,)`
C Element	`siccd`	FORTRAN Element	`SICCD`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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	Header Identification and Summary Data Header Identification and Data Range Variables		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`/hh, /hr, /hrl`
C Object	`header_row`	FORTRAN Common Block	`/HEADER/`
C Array	`header`	FORTRAN Array	`n/a`
C Element	`hsiccd`	FORTRAN Variable	`HSICCD`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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*.

Primary Concept	Index Rebalancing History Arrays		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`rebal_ts[]`	FORTRAN Common Block	`n/a`
C Array	`rebal[][]`	FORTRAN Array	`n/a`
C Element	`maxid`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`IND`

Statistic Maximum in Period

Maximum Statistic in Period is a maximum statistic value in the portfolio at the beginning of a rebalancing period.

Primary Concept	Index Rebalancing History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	rebal_ts[]	FORTRAN Common Block	n/a
C Array	rebal[][]	FORTRAN Array	n/a
C Element	maxstat	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Statistic Median in Period

Median Statistic in Period is the median statistic value in a portfolio at the beginning of a rebalancing period.

Primary Concept	Index Rebalancing History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	rebal_ts[]	FORTRAN Common Block	n/a
C Array	rebal[][]	FORTRAN Array	n/a
C Element	medstat	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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*.

Primary Concept	Index Rebalancing History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	rebal_ts[]	FORTRAN Common Block	n/a
C Array	rebal[][]	FORTRAN Array	n/a
C Element	minid	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Statistic Minimum in Period

Minimum Statistic in Period is the minimum statistic value in the portfolio at the beginning of the rebalancing period.

Primary Concept	Index Rebalancing History Array		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	rebal_ts[]	FORTRAN Common Block	n/a
C Array	rebal[][]	FORTRAN Array	n/a
C Element	minstat	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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. This table describes the suffixes appearing on the CRSP file.

Nasdaq 5th 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 comprise 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 1970's through the early 1980's.

NYSE tickers prior to July 1962 are blank.

Primary Concept	Name History Array		
ts_print Daily Usage	`ticker/0`		
ts_print Monthly Usage	`mticker/0`	*stk_print* Option(s)	`/n`
C Object	`names_arr`	FORTRAN Common Block	`/INFO/`
C Array	`names[]`	FORTRAN Array	`CNAMES(,) or TICK function`
C Element	`ticker`	FORTRAN Element	`TICKER,`
Database Format(s)		Data Type(s)	`STK`

Ticker Symbol, Header

Ticker Code, Header is the ticker on the current exchange. It is only set if the company is active at the end of the file and is blank otherwise. If the security trades on NYSE or AMEX and is distinguished with a non-blank share class, Ticker Code, Header includes the most current Ticker Symbol followed by a period and the *Share Class*. Ticker Symbol, Header can range from one to five characters long. Tickers and Share Classes are derived from the most recent information in the name history.

Primary Concept	Header Identification and Summary Data		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	/hh, /hr, /hrl
C Object	header_row	FORTRAN Common Block	n/a
C Array	header	FORTRAN Array	n/a
C Element	htick	FORTRAN Variable	n/a
Database Format(s)	CA97	Data Type(s)	STK

Total Count of Market

Total Count of Market is the number of stocks in the market defined in a market index with a valid price on the selected *Calendar Trading Date*.

Primary Concept	Calendar/Indices, Decile Returns, Index Levels Arrays		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAL/
C Array	n/a	FORTRAN Array	TOTCNT()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	STK, IND

Total Value of Market

Total Value of Market contains the total market value for a given market, in $1000's, for all non-ADR securities in the market defined in a market index with valid prices on the selected *Calendar Trading Date*.

Primary Concept	Stock File Indices, Index Levels Arrays		
ts_print Daily Usage	cap/0		
ts_print Monthly Usage	mcap/0	*stk_print* Option(s)	n/a
C Object	n/a	FORTRAN Common Block	/CAL/
C Array	n/a	FORTRAN Array	TOTVAL()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	STK, IND

Universe Subset Type Code (in a 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, ADR, REIT, Closed End Funds
11	NYSE/AMEX common excluding foreign, ADR, REIT, Closed End Funds
12	NYSE/AMEX/The Nasdaq National Market common excluding foreign, ADR's, 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

Primary Concept	Index Header		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`indhdr_row`	FORTRAN Common Block	`n/a`
C Array	`partuniv[]` `induniv[]`	FORTRAN Array	`n/a`
C Element	`partunivcode` `indunivcode`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`IND`

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 use in the actual index. The following base codes are used. The sum of two or more codes indicates all selected exchanges are valid.

Code	Description
0	No exchange restriction
1	New York Stock Exchange
2	American Stock Exchange
4	Nasdaq Stock Market

Primary Concept	Index Header		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`indhdr_row`	FORTRAN Common Block	`n/a`
C Array	`partuniv[]` `induniv[]`	FORTRAN Array	`n/a`
C Element	`wantexch`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`IND`

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 n^{th} 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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	indhdr	FORTRAN Array	n/a
C Element	fstdig	FORTRAN Element	n/a
Database Format(s)		Data Type(s)	IND

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 acutal 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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	partuniv[] induniv[]	FORTRAN Array	n/a
C Element	wantinc	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	partuniv[] induniv[]	FORTRAN Array	n/a
C Element	wantnms	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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 acutal index. Valid Second Digit of *Share Code* is the decimal representation of a 10-digit binary number. The n^{th} 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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	partuniv[] induniv[]	FORTRAN Array	n/a
C Element	secdig	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

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	Index Header		
ts_print Daily Usage	n/a		
ts_print Monthly Usage	n/a	*stk_print* Option(s)	n/a
C Object	indhdr_row	FORTRAN Common Block	n/a
C Array	partuniv[] induniv[]	FORTRAN Array	n/a
C Element	wantwi	FORTRAN Element	n/a
Database Format(s)	CA97	Data Type(s)	IND

Volume Traded

Volume Traded is the integer raw number of shares traded during the calendar period. It is expressed in units of one share, although 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 paper trades are included in the volume.

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

daily: Volume Traded is the total number of shares of a stock traded on that day.

monthly: Volume Traded is the sum of the trading volumes during that month. Monthly volumes are the sum of raw shares and are not adjusted for splits during the month.

Primary Concept	Price, Volume and Return Arrays		
ts_print Daily Usage	`vol/0`		
ts_print Monthly Usage	`mvol/0`	*stk_print* Option(s)	`/pv, /dd`
C Object	`vol_ts`	FORTRAN Common Block	`/DDATA/`
C Array	`vol[]`	FORTRAN Array	`VOL()`
C Element	`n/a`	FORTRAN Element	`n/a`
Database Format(s)	`CA97, SFA`	Data Type(s)	`STK`

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	Index List History Array		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`
C Object	`list_arr[]`	FORTRAN Common Block	`n/a`
C Array	`list[][]`	FORTRAN Array	`n/a`
C Element	`weight`	FORTRAN Element	`n/a`
Database Format(s)	`CA97`	Data Type(s)	`IND`

Year and Month of Quarter

Year and Month of Quarter contains the year and month of the quarter used to define the rebalancing period of a Cap-Based Portfolio. A quarter is labeled by the last month in the quarter. The CRSP quarterly calendar used with Cap-Based rebalancing histories begins in March, 1926, the end of the first quarter containing results data rather than at the inception of the file, December 31, 1925. Rebalancing is based on data at the end of the previous quarter.

Primary Concept	Cap-Based Reports Rebalancing History Arrays		
ts_print Daily Usage	`n/a`		
ts_print Monthly Usage	`n/a`	*stk_print* Option(s)	`n/a`

C Object	n/a	FORTRAN Common Block	/REBAL/
C Array	n/a	FORTRAN Array	YYYYMM()
C Element	n/a	FORTRAN Element	n/a
Database Format(s)	CA97, SFA	Data Type(s)	IND

CHAPTER 5: CRSP CALCULATIONS

OVERVIEW

This chapter contains calculations used in the CRSP stock and indices databases, organized alphabetically by name.

INSIDE

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

Adjusted data is price, dividend, shares, and volume data historically revised for split events to make data directly comparable at different times during the history of a security. CRSP provides unadjusted data, but data utilities can be used to generate adjusted data.

An adjustment base date is chosen as the anchor date. All data on this date is unadjusted, and other data is 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.

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 no split events before $t+1$,	$C(t) = C(t+1)$
if $t < C_0$ and a split event with factor f before $t+1$,	$C(t) = C(t+1) / f$

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.

Associated Portfolio Returns

Associated portfolio returns are a composite of a group of portfolio index series based on security and time-dependent portfolio assignments. 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 that the security belongs to 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 Capital Appreciation contains returns without dividends for a security over a holding period. It is similar to Total Returns except ordinary dividends and certain other regularly taxable dividends are excluded from the returns calculation. The formula is the same as for Security Total Returns except *d(t)* is usually 0. See Total Returns for missing values. Capital Appreciation is also known as returns 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; so is 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. 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 the 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 are 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. In monthly files a 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.

When valuing a portfolio, the delisting return or other representation can be used to assign a value to the delisted security. The user must decide whether to assign alternate estimated values based on the Delist Code when delisting payment information is unavailable.

Dividend Amount

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. 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.

Factor to Adjust Prices in Period

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 Prices 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 prices 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 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 returns, the security returns and the index returns are cumulated separately before subtracting the difference.

CRSP Beta Excess Returns and Standard Deviation Excess returns are special cases of excess returns previously available on CRSP files. These are based on the NYSE/AMEX Beta and Standard Deviation Portfolios and the Nasdaq Beta and Standard Deviation Portfolios. The NYSE/AMEX Beta Excess returns are calculated using the Trade-Only Returns of the NYSE/AMEX securities.

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 returns 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 also called 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 Security Total Returns for missing values.

Index Count

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

Index Levels

An 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 $t = D_0$	$i_t = V_0$
if $t > D_0$	$i_t = i_{t-1}(1+r_t)$
if $t < D_0$ and $t \geq t_0$	$i_t = i_{t+1} / (1+r_{t+1})$
if $t < t_0$	$i_t = 0$

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 indexes 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 array.

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 of the portfolio $V(I) = \sum_n v_n(I) = \sum_n w_n(I)$

Market Capitalization

Market Capitalization 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* variable and the applicable shares observation from the Shares Outstanding Observation Array for each calendar period to calculate Market Capitalization.

Rebasing Index Levels

It is possible to rebase an index to make index levels to 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 return 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 period 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. 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 price in period is derived from the distribution history Factor to Adjust Price 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 Cash Dividend Amount 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, Price Adjustment Factor is 2.0, and Cash Adjustment is 0.0. If a one dollar dividend is the only distribution event in the time range, both Cash Dividend Amount and Cash Adjustment are 1.0.

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

Missing Return Codes

Holding Period Total Return (*t*)	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	outside the range of the security's price range
-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

$$\beta_i = \frac{\sum_t (ret_{i,t}\ mret3_t) - \left(\frac{1}{n}\right)\left(\sum_t ret_{i,t}\right)\left(\sum_t mret3_t\right)}{\sum_t (mret_t\ mret3_t) - \left(\frac{1}{n}\right)\left(\sum_t mret_t\right)\left(\sum_t mret3_t\right)}$$

where summations over t are over all days on which security i traded, beginning with the second trading day of the year and ending with the second to 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 nonmissing 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 Small Cap securities after June 15, 1992. Nasdaq returns based on bid/ask averages have different characteristics than 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 asked) of security i on day t.
n = number of observations for the year (of $ret_{i,t}$)
$\sigma_i \equiv$ yearly standard deviation for the i^{th} company

$$\sigma_i = \frac{\sqrt{\sum_t (ret_{i,t})^2 - \frac{1}{n}\left(\sum_t ret_{i,t}\right)^2}}{n-1}$$

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 Closing Price or Bid/Ask average 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 is provided unadjusted. However, the distribution history can be used to generate adjusted data 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 CODING SCHEMES

OVERVIEW

This chapter contains basic coding systems used in the CRSP stock and indices data.

INSIDE

CHAPTER 6. CRSP DATA CODING SCHEMES

6.1 Name History Array Codes

CINS Country Codes used in CUSIP

A = Austria	B = Belgium	C = Canada	D = Germany
E = Spain	F = France	G = United Kingdom	H = Switzerland
J = Japan	K = Denmark	L = Luxembourg	M = Mid-East
N = Netherlands	P = South America	Q = Australia	R = Norway
S = South Africa	T = Italy	U = United States	V = Africa - Other
W = Sweden	X = Europe Other	Y = Asia	

Share Type

This table lists the share type codes found in the CRSP stock files. The first digit describes the type of security traded.

Share Type - Digit #1 - Security Traded

Code	Definition
1	Ordinary Common Shares
2	Certificates
3	ADRs (American Depository Receipts)
4	SBIs (Shares Of Beneficial Interest)
7	Units (Depository Units, Units Of Beneficial Interest, Units Of Limited Partnership Interest, Depository Receipts, Etc.)

The second digit describes more detailed information about the type of security.

Share Type - Digit #2 - 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 (Primes And Scores)
4	Closed-End Funds
5	Closed-End Fund Companies Incorporated Outside The U.S.
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 NYSE Or AMEX
-1	Suspended By NYSE Or AMEX
0	Not Trading On NYSE, AMEX, Or The NASDAQ Stock Market
1	New York Stock Exchange
2	American Stock Exchange
3	The NASDAQ Stock Market
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 The New York Stock Exchange
32	When-Issued Trading On The American Stock Exchange
33	When-Issued Trading On NASDAQ

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 as 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.

Distribution Codes

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
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 digit 1 equals “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 digit 1 equals “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 digit 1 = "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 digit 1 equals “4” Only)	0	unknown, not yet coded
	1	unspecified or not applicable
	2	market value
	3	indicated value
3 Split Type (for digit 1 equals “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

Digit	Code	Meaning
	6	distribution of different issue of common; same company
	7	initial distribution of other class of common; same company
3 Types of Offer or Reason for Issuance (for digit 1 equals "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 digit 1 equals "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 receipt
	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 4th 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.

Distribution Events Table

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.
	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.

Category	Code	Description
Dividend *(con't)*	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 receipt.
	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.
	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 merger or exchange. Amount or some terms missing. Fully taxable as ordinary income to individuals
	3989*	Debenture without established market value, tax status - dividend reinvestment plan qualifies for the limited exclusion provided by Sec. 305(e) of the Internal Revenue Code.
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.

Category	Code	Description
Rights *(Con't)*	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/decreased by merger with company not on file, tax status - gain or loss realized compared with cost.
	6261	Common shares increased/decreased through a companies own tender offer, tax status - unknown.
	6511	Common shares increased for reasons not specified.
	6521	Common shares decreased 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.
	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 uncoded information 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
	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
	205	When merged, shareholders receive shares of mutual funds
	240*	Flags merger with missing final distribution information
Exchanges	300	Issue acquired by exchange of stock
	301	Issue exchanged for issue trading on NYSE
	302	Issue exchanged for issue trading on AMEX
	303	Issue exchanged for issue trading on NASDAQ
	320	Issue exchanged for stock trading Over-the-Counter
	340*	Flags an exchange with missing final distribution information
	350*	Flags an exchange attempt that was not sufficient to "kill" issue
	390*	Flags an unsuccessful exchange attempt with missing distribution information
Liquidations	400	Issue stopped trading as result of company liquidation
	401	Issue liquidated, for issue trading on NYSE
	403	Issue liquidated for issue trading on Nasdaq
	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.
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
	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
	535	Unlisted trading privileges. Removed by Exchange.
	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
	590	Delisted by current exchange - underlying assets have merged with another company
	600	Expired warrant or right
	601	Warrants, rights, or units called for redemption
	610	Unit split into its component parts
	700	Issue Delisted by Securities Exchange Commission
	801*	Issue Simultaneously listed on NASDAQ and NYSE
	802*	Issue Simultaneously listed on NASDAQ and AMEX

*These codes are intended to alert the user to delisting events undergoing further research. The individual digits in these codes do *not* necessarily conform to CRSP's standard delisting coding system.

6.4 NASDAQ Information Codes

Trading Status Code

TRTSCD	Description
0	unknown
1	active
2	only one market maker
3	suspended
4	inactive
5	delisted

NASDAQ National Market (NMS) Indicator Code

NMSIND	Description
0	unknown
1	NASDAQ Small-Cap before June 15, 1992
2	NASDAQ National Market
3	NASDAQ Small-Cap after June 15, 1992

Nasdaq Index Code

NMSIND	Description
0	unknown
1	NASDAQ Small-Cap before June 15, 1992
2	NASDAQ National Market
3	NASDAQ Small-Cap after June 15, 1992

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 return, array index not within range of BEGRET and ENDRET
RMISSP	-99.0	missing return due to missing price

APPENDIX

INSIDE

Appendix

APPENDIX A: 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.

APPENDIX B: 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 indexes 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 only contains the dates where trading was conducted on a major US exchange. Data is 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 CRSPAccess97 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 is always associated with one of these calendars. The list of time series observations are 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 CRSPAccess97 databases are:

Calendars	Calendar Identification Number	Calendar Name	Beginning Date
daily	100	Daily Trading Calendar	19620702
monthly	101	Month-end Trading Calendar	19251231
annual	300	Annual Trading Calendar	19251231
quarterly	310	Quarterly Trading Calendar	19251231
weekly	500	Weekly Trading Calendar	19620706

Calendar/Indices

In the CRSPAccess97 FORTRAN sample programming usage and the CRSP SFA format, 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-weighed index, an equal-weighted index, and a composite index. In this format the calendar/indices calendar is applicable to all time series except portfolios in the corresponding stock file.

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 depends 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.

CRSPAccess97 Format

CRSPAccess97 is the name given to the format of CRSP stock and indices data files introduced in 1996. The CRSPAccess97 format is a binary format with data utilities and C and FORTRAN random access data libraries supported of 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 CRSPAccess97 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 CRSPAccess97 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 events, observations, 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. The status changes and observations usually contain information that is in effect until modified by another similar event. The second type records events as they occur.

Examples are event arrays in CRSP Stock files are the Names 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 includes the permanent security identifier, the header CUSIP, and date ranges of a security. CRSP indices file header data includes index identifier and flags describing the index methodology.

See Data Objects, CRSPAccess97 Programmers Guide FORTRAN `/HEADER/` common block, C `CRSP_ROW`, stock `header` array, and indices `indhdr` array.

Index Groups

An 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 provide convenient access when comparing a security whose membership may fluctuate from portfolio to portfolio over time.

Index groups are only available in the CRSP US Stock, Treasury Indices and Portfolio Assignments Database.

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.

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 Small-Cap Market.

Non-Ordinary Dividends

Non-Ordinary dividends are distributions made to shareholders of a security representing the return of capital. These are factored into the capital appreciation of the security.

Ordinary Dividends

Ordinary Dividends 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.

PERMCO

The CRSP PERMCO is an integer used to uniquely define all companies 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.

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. A unique CUSIP issuer number is only found in the CRSP *PERMNO*.

INDNO

In CRSPAccess97 databases, all indexes are assigned a Permanent Index Identification Number, `INDNO`. See the Indices Methodologies section for a full list of available CRSP indices and information on the composition and methodologies for calculating the different indices.

PERMNO

The PERMNO is an integer used to uniquely define all securities in a CRSP stock database. The PERMNO does not change historically if the security changes names or makes capital changes and 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

A portfolio type is an integer identifying a specific defined portfolio methodology within a CRSPAccess97 stock database set. 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 CRSPAccess97 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 CRSPAccess97 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 CRSPAccess97 database.

The predefined Set Identifiers in CRSPAccess97 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 CRSPAccess97 database. Stock databases support stock (STK), index (IND), and calendar (CAL) set types. Data for each set type has the same data variables, identifiers, and CRSP programming library access functions.

SFA Format

SFA Format is the name given to CRSP data files with defined character and binary record layouts and provided with sequential FORTRAN access sample programs. This is the format of CRSP files provided on tape with SAS `Proc Datasource` support, including modifications for year-2000 compliance. SFA Format files can be generated from CRSPAccess97 databases using utility programs provided.

See *The CRSP SFA Guide* for complete information on the SFA format.

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 does not meet the selected restrictions is 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.

D

E

F

G

H

I

O

P

R

S

T

U

V

W

Y