

Running PC-SPAN Reports and Export Datafiles in Batch

Often, PC-SPAN users would like to issue one command which would accomplish some or all of the following:

- Load one or more SPAN risk parameter files
- Load a position file
- Calculate performance bond requirements
- Save the results to a SPAN document file
- Create reports showing the performance bond requirements for each portfolio and the positions in each portfolio
- Create reports showing product data and risk array data
- Create a datafile from which the performance bond requirement data can be imported into other application(s)
- Create datafile(s) which allows product data to be imported into other applications.

This type of task can be accomplished using several utilities included with PC-SPAN -- **spanit**, **spanReport**, **spanProducts**, and **spanProductsReport**.

- [Using the spanReport Utility](#)
- [Output of the spanReport Utility](#)
- [Using the spanProducts Utility](#)
- [Output of the spanProducts Utility](#)
- [Using the spanProductsReport Utility](#)
- [Output of the spanProductsReport Utility](#)
- [Logging](#)
- [Contents of the Datafiles Produced by spanProducts](#)

The **spanit** utility allows you to create a simple script file which accomplishes loading of SPAN file(s), loading of position file(s), calculation of performance bond requirements, and saving the results to an XML-based SPAN document file (a **.spn** file).

The **spanReport** utility allows you to take that SPAN document file and in one operation generate three reports:

- The **Performance Bond Requirements** report, detailing the portfolios, their calculated requirements, and the breakdown of those requirements
- The **Positions** report, listing all of the positions in each portfolio, and
- The **Performance Bond Requirements Datafile**, containing the performance bond requirements in a comma-delimited format (a **.csv** - "comma separated values" - file), which can be imported into spreadsheet programs and other applications.

The **spanProducts** utility similarly allows you to take the SPAN document file as input, and in one operation generate four comma-delimited datafiles which can then easily be imported into other applications:

- The **Products Datafile**, containing the definition of all product families
- The **Futures-Physicals-Combinations Datafile**, providing data about all specific products which are not options
- The **Option Series Datafile**, providing information about each option series -- *ie*, each set of options with the same expiration and the same underlying, and
- The **Options Datafile**, providing information about each option.

The **spanProductsReport** utility allows you to take the SPAN document file as input and in one operation generate four reports:

- the **Products** report, listing product families by combined commodity
- the **Contracts** report, listing contracts (tradeable instruments) together with their prices, volatilities, times to expiration, etc.
- the **Risk Arrays** report, listing prices and risk arrays for contracts, and
- the **Risk Arrays - Standard** report, listing prices, risk arrays showing the "standard" 16 risk scenarios, and formatting the report to show risk array values in ascending order by underlying price movement.

The **spanReport**, **spanProducts**, and **spanProductsReport** utilities are all examples of PC-SPAN's batch reporting capability, and their use is described herein.

The executions of **spanit**, **spanReport**, **spanProducts**, and/or **spanProductsReport** can themselves be accomplished from within a batch file. By creating such a batch file, and by placing a shortcut to this batch file on the Windows desktop, you can initiate entire production processes with one mouseclick.

Using the **spanReport** Utility

The full name of this utility is **spanReport.hta** and it is located in the **rptModule** subdirectory of the directory where PC-SPAN is installed. Typically this would be **c:\span4\rptModule**. The **.hta** stands for "hyper-text application."

You can run the utility either (a) from within the **rptModule** directory, or (b) by providing the path to the **rptModule** directory. The syntax is:

```
\span4\rptModule\spanReport.hta xmlFileName [busDate] [firm]
```

where

- **xmlFileName** is a SPAN document file name. If the full path to this SPAN document file is not specified, the file is assumed to be located in **\span4\data**.
- **busDate** is a business date in a CCYYMMDD format, e.g. 19990423. If the date is not specified or there are more than one 'pointInTime' nodes for the same date, reports will be generated for the first 'pointInTime' node in the file. If there is more than one point in time for the same business date, reports will be generated for the first point in time for that date.

- *firm* is the clearing member firm or other firm identifier for a portfolio. If provided, the generated reports will include data for all accounts defined for the specified firm. If it is not specified, reports will be generated for all portfolios in the file.

For example, suppose that the SPAN document file **span.spn** contains results for a single business date, November 11, 2000. The following command would generate the three reports for all portfolios for that business day.

```
spanReport.hta span.spn
```

The following command would generate the three reports for all accounts for firm **001** for that business day:

```
spanReport.hta span.spn 20001111 001
```

If **span.spn** contained data for November 11th and November 12th, the following command would generate reports for all accounts for the 12th:

```
spanReport.hta span.spn 20001112
```

and the following command would generate reports for all accounts for firm **001** for the 12th:

```
spanReport.hta span.spn 20001112 001
```

Output of the spanReport Utility

When you run **spanReport.hta**, if the reports could be successfully generated, the results are created as three files in the **Reports** subdirectory of the application directory, typically **\span4\Reports**. The files are:

- **Positions.htm**
- **PB Requirements.htm**
- **PB Req Delim.txt**

The first two files can then be opened in any Web browser (Microsoft Internet Explorer version 5.5 recommended) for viewing and/or printing.

The third file, containing performance bond requirements in comma-delimited format, can be imported into other applications.

Using the spanProducts Utility

Using **spanProducts** is almost exactly analogous to using **spanReport**. The full name of the utility is **spanProducts.hta** and it is located in the **rptModule** subdirectory of the directory where PC-SPAN is installed. Typically this would be **c:\span4\rptModule**.

You can run the utility either (a) from within the **rptModule** directory, or (b) by providing the path to the **rptModule** directory. The syntax is:

```
\span4\rptModule\spanProducts.hta xmlFileName [busDate]
```

where

- *xmlFileName* is a SPAN document file name. If the full path to this SPAN document file is not specified, the file is assumed to be located in **\span4\data**.
- *busDate* is a business date in a CCYYMMDD format, e.g. 19990423. If the date is not specified or there are more than one 'pointInTime' nodes for the same date, reports will be generated for the first 'pointInTime' node in the file. If there is more than one point in time for the same business date, reports will be generated for the first point in time for that date.

Output of the spanProducts Utility

When you run **spanProducts.hta**, if the datafiles could be successfully generated, the results are created as four files in the **Reports** subdirectory of the application directory, typically **\span4\Reports**. The files are:

- **Products.txt**
- **Futures-Physicals-Combinations.txt**
- **Series.txt**
- **Options.txt**

Using the spanProductsReport Utility

The full name of the utility is **spanProductsReport.hta** and it is located in the **rptModule** subdirectory of the directory where PC-SPAN is installed. As with the other utilities, you can run it either (a) from within the **rptModule** directory, or (b) by providing the path to the **rptModule** directory. The syntax is:

```
\span4\rptModule\spanProductsReport.hta xmlFileName [busDate [ec [cc] ]]
```

where

- *xmlFileName* is a SPAN document file name. If the full path to this SPAN document file is not specified, the file is assumed to be located in `\span4\data`.
- *busDate* is a business date in a CCYYMMDD format, e.g. 19990423. If the date is not specified or there are more than one 'pointInTime' nodes for the same date, reports will be generated for the first 'pointInTime' node in the file. If there is more than one point in time for the same business date, reports will be generated for the first point in time for that date.
- *ec* is an exchange complex acronym
- *cc* is a combined commodity code

If the exchange complex acronym is not provided, the report will be produced for all exchange complexes for the specified point in time. Similarly, if a combined commodity code is not specified, the report will be produced for all combined commodities for the specified exchange complex.

Output of the spanProductsReport Utility

When you run **spanProductsReport.hta**, if the reports could be successfully generated, the results are created as various files in the **Reports** subdirectory of the application directory, typically `\span4\Reports`. The files are:

- **Products<xx>.htm**
- **Contracts<xx>.htm**
- **RiskArrays<xx>.htm**
- **RiskArraysStd<xx>.htm**

where **<xx>** identifies the exchange complex and combined commodity for which the report was generated.

Logging

A log file is also created in the **Reports** subdirectory, named **rptBatch.log**. Any previous version of this logfile is overwritten. The following is an example of a log file:

rptBatch.log

Batch Report Utility log

8/30/2000 9:59

Batch Parameters:

XML File: ../data/mult_pit.spn

Bus Date: undefined

Firm: undefined

Generated 'Positions' report

Generated 'PB Requirements' report

Generated 'PB Req Delim' report

*** Normal completion ***

However, if errors were detected, such that the reports cannot be created, the utility generates an error log that describes what went wrong. The error log is also created in the **Reports** subdirectory and is named **rptBatch.err**. (Any previous version of the error log is overwritten.) The following is an example of an error log file:

rptBatch.err

Batch Report Utility log

8/30/2000 9:57

*** Operation failed !!! ***

ERROR: Not all required parameters are specified

Usage: rptBatch.hta xmlFileName [busDate] [firm]

Contents of the Datafiles Produced by spanProducts

As described above, **spanProducts** produces four comma-delimited datafiles:

- **Products.txt**
- **Futures-Physicals-Combinations.txt**
- **Series.txt**
- **Options.txt**

Each of these begins with a line of column-headers, identifying each data element. Here are the detailed descriptions of each file and its data elements.

Products.txt: these contains a record for every **product family** for the selected point in time. For example, if CME data is loaded, there would be records for product families including S&P 500 futures, options on S&P 500 futures, E-Mini S&P 500 futures, and options on E-Mini S&P 500 futures. Specific data elements are:

- **date** - business date
- **isSettl** - **1** if this point in time pertains to the end-of-day settlement, or **0** if this is for an intraday point in time
- **qual** - qualifier for that point in time. For example, **final** for the final end-of-day settlement, **early** for an early (or preliminary) end-of-day settlement, or **complete** for an end-of-day settlement containing additional data not available in the early and final files
- **time** - business time associated with the point in time, primarily for use with intraday points in time
- **run** - run number, allowing further qualification of the point in time
- **ec** - exchange complex acronym -- either the clearing organization acronym or the cross-margin agreement or other business function acronym
- **cc** - combined commodity code
- **exch** - exchange (or exchange group) acronym
- **pfType** - identifies the product type, for example **FUT** for futures, **OOF** for option on future
- **pfld** - ID number of this product family
- **pfCode** - the primary product code used for this product family, often called the "commodity code"
- **name** - product family name
- **undPfType** - product type of the product family underlying of this product family. Not defined for physical product families or combination product families.
- **undPfld** - the ID number of the underlying product family.
- **undPfCode** - the product code of the underlying product family
- **currency** - the ISO currency code for this product family's settlement currency
- **stdCvf** - the standard contract value factor for the product family, *ie*, the multiplier which converts a quoted price to a contract value
- **priceDI** - price decimal locator -- the decimal locator for settlement prices provided in the 7-digit format
- **priceFmt** - the "alignment code" for settlement prices, used to denote a particular nondecimal price format. Blank for decimal prices.
- **valueMeth** - the "valuation method" - **FUT** or **EQTY** to denote a product valued futures-style or a product valued premium-style ("equity-style")
- **priceMeth** - the price quotation method
- **setlMeth** - the settlement method for this product family - **CASH** for a cash-settled future, **DELIV** for a physically deliverable future and for all options (which are settled by delivering the underlying, whatever it may be)
- **strikeDI** - the decimal locator for strike prices provided in the 6-digit format
- **strikeFmt** - the alignment code for strike prices, used to denote particular non-decimal formats. Blank for decimal strike prices.
- **cab** - cabinet option value -- for option products, the monetary value to be assigned to "fixed" cabinet prices, if this concept is applicable

Futures-Physicals-Combinations.txt: contains a record for every tradeable instrument at the specified point in time which is not an option. Specific data elements are as shown. Descriptions are provided only for those elements which are not present in the **Products** datafile.

- **date** - business date
- **isSettl** - **1** if this point in time pertains to the end-of-day settlement, or **0** if this is for an intraday point in time
- **qual** - qualifier for that point in time. For end-of-day settlements, **final**, **early** or **complete**
- **time** - business time, primarily for intraday points in time
- **run** - run number
- **ec** - exchange complex acronym -- clearing organization acronym, cross-margin agreement, or other business function acronym
- **cc** - combined commodity code
- **exch** - exchange (or exchange group) acronym
- **pfType** - product type, for example **FUT** for futures, **OOF** for option on future
- **pfld** - ID number of this product family
- **pfCode** - primary product code for this product family
- **clid** - contract ID number - specific ID number of this particular tradeable instrument
- **pe** - period code of this contract
- **undPfType** - product type of the product family underlying of this product family. Not defined for physical product families or combination product families.
- **undPfld** - the ID number of the underlying product family. Not defined for physicals or combinations.
- **undPfCode** - the product code of the underlying product family. Not defined for physicals or combinations.
- **undClid** - underlying contract ID number. Not defined for physicals or combinations.
- **undPe** - period code of the underlying contract. Not defined for physicals or combinations.
- **currency** - the ISO currency code for this product family's settlement currency
- **stdCvf** - standard contract value factor for the product family.
- **p** - contract price
- **value** - contract monetary value in the settlement currency
- **cvf** - contract value factor for this specific product, which may be different from the standard contract value factor
- **sc** - scaling factor associated with this product
- **setlDate** - the settlement date for this product, defined as the date on which the final price is determined
- **t** - time to maturity, in years, from the specified point in time to the settlement date
- **r** - the rate ID for the **priceScan** and **volScan** values to follow, a integer denoting the account type, performance bond class, and initial /maintenance designation to which these rates pertain
- **priceScan** - the price scan range for the specified rate ID, in the performance bond currency
- **volScan** - the volatility scan range for the specified rate ID, as an absolute decimal fraction
- **d** - SPAN composite delta for the specified rate ID
- **a1** through **a16** - risk array values for the standard 16 risk scenarios for the specified rate ID

Series.txt: contains a record for every option series for the specified point in time. Descriptions are provided only for those elements which are not present in the **Futures-Physicals-Combinations** datafile. Specific data elements are:

- **date** - business date
- **isSettl** - **1** if this point in time pertains to the end-of-day settlement, or **0** if this is for an intraday point in time

- **qual** - qualifier for that point in time. For end-of-day settlements, **final**, **early** or **complete**
- **time** - business time, primarily for intraday points in time
- **run** - run number
- **ec** - exchange complex acronym -- clearing organization acronym, cross-margin agreement, or other business function acronym
- **cc** - combined commodity code
- **exch** - exchange (or exchange group) acronym
- **pfType** - product type, for example **FUT** for futures, **OOF** for option on future
- **pfld** - ID number of this product family
- **pfCode** - primary product code for this product family
- **cld** - contract ID number - not defined for an option series
- **pe** - period code
- **undPftType** - product type of the product family underlying of this product family.
- **undPfld** - the ID number of the underlying product family.
- **undPfCode** - the product code of the underlying product family.
- **undCld** - underlying contract ID number
- **undPe** - period code of the underlying contract.
- **currency** - the ISO currency code for this product family's settlement currency
- **stdCvf** - standard contract value factor for the product family.
- **p** - contract price (not defined for an option series)
- **value** - contract monetary value in the settlement currency (not defined for an option series)
- **cvf** - contract value factor for this series of options, which may be different from the standard contract value factor for the product family.
- **sc** - scaling factor associated with this series
- **setlDate** - the settlement date for this option series, defined as the date on which the final price is determined. Synonymous with expiration date for the option series.
- **t** - time to maturity, in years, from the specified point in time to the settlement date
- **intrRate** - interest rate, as a decimal fraction. Typically the annualized risk-free continuously-compounded rate.
- **divRate** - dividend yield, as a decimal fraction. Typically the annualized continuously-compounded rate
- **r** - the rate ID for the **priceScan** and **volScan** values to follow, a integer denoting the account type, performance bond class, and initial /maintenance designation to which these rates pertain
- **priceScan** - the price scan range for the specified rate ID, in the performance bond currency
- **volScan** - the volatility scan range for the specified rate ID, as an absolute decimal fraction

Options.txt: contains a record for every option for the specified point in time. Descriptions are provided only for those elements which are not present in the **Series** datafile. Data elements are:

- **date** - business date
- **isSetl** - 1 if this point in time pertains to the end-of-day settlement, or 0 if this is for an intraday point in time
- **qual** - qualifier for that point in time. For end-of-day settlements, **final**, **early** or **complete**
- **time** - business time, primarily for intraday points in time
- **run** - run number
- **ec** - exchange complex acronym -- clearing organization acronym, cross-margin agreement, or other business function acronym
- **cc** - combined commodity code
- **exch** - exchange (or exchange group) acronym
- **pfType** - product type, for example **FUT** for futures, **OOF** for option on future
- **pfld** - ID number of this product family
- **pfCode** - primary product code for this product family
- **cld** - contract ID number
- **pe** - period code
- **o** - option right code - **C** for call or **P** for put
- **k** - strike price
- **undPftType** - product type of the product family underlying of this product family. Not defined for physical product families or combination product families.
- **undPfld** - the ID number of the underlying product family. Not defined for physicals or combinations.
- **undPfCode** - the product code of the underlying product family. Not defined for physicals or combinations.
- **undCld** - underlying contract ID number
- **undPe** - period code of the underlying contract.
- **currency** - the ISO currency code for this product family's settlement currency
- **stdCvf** - standard contract value factor for the product family.
- **p** - contract price
- **value** - contract monetary value in the settlement currency
- **cvf** - contract value factor for series containing this option, which may be different from the standard contract value factor for the product family
- **sc** - scaling factor associated with the series containing this option
- **setlDate** - the settlement date for series containing this option, defined as the date on which the final price is determined. Synonymous with expiration date.
- **t** - time to maturity, in years, from the specified point in time to the settlement date
- **intrRate** - interest rate, as a decimal fraction. Typically the annualized risk-free continuously-compounded rate.
- **divRate** - dividend yield, as a decimal fraction. Typically the annualized continuously-compounded rate
- **d** - SPAN composite delta for the specified rate ID
- **v** - implied volatility for this option, as a decimal fraction
- **r** - the rate ID for the **priceScan** and **volScan** values to follow, a integer denoting the account type, performance bond class, and initial /maintenance designation to which these rates pertain
- **priceScan** - the price scan range for the specified rate ID, in the performance bond currency
- **volScan** - the volatility scan range for the specified rate ID, as an absolute decimal fraction
- **d** - SPAN composite delta for the specified rate ID
- **a1** through **a16** - risk array values for the standard 16 risk scenarios for the specified rate ID