Market Quality Dashboard – Metrics Design Functional Specification

Authors: Shan Ji

Version: 1.0

Document Update History

Date	Version	Change
13/08/2013	1.0	Shan Ji create version 1.0

MQ Dashboard Metrics

Metric 1: On Market Volume Share by Venue 4
Metric 2: Daily Percentage of NBBO Time by Venue 5
Metric 3: Order to Trade Ratio 7
Metric 4 & 18: Effective Spread (bps) and Realised Spread (bps) 8
Metric 5: Intraday Volatility 1 10
Metric 6: Info Leakage 11
Metric 7: On Market Value by Venue 15
Metric 8: On Market Trade Count by Venue 16
Metric 9: On Market Square-root Value by Venue 17
Metric 10: Scope by Venue 18
Metric 11: Daily Percentage of Tightest Spread Time by Venue 19
Metric 12: Daily Percentage of NBBO Depth (dollar value) by Venue 21
Metric 13: On Market Trade Size by Venue 23
Metric 14: Quoted Value near the BBA (within 50bps) 24
Metric 15: Number of Quote Updates 26
Metric 16 & 17 - Quoted Spread ($) and Quoted Spread (bps) 27
Metric 21: Intraday Volatility 2 29
Metric 22: Interday Close Price Volatility 30
Metric 23: Quote Stuffing 31
Metric 24: Dislocation of EOD Price 33
Metric 25: Market Resiliency 37
Metric 26: Price Discovery 39

MQ Dashboard Metrics

Metric 1: On Market Volume Share by Venue

Event types	On-market trade
Data Source	TRTH Quotes and Trades database for local ATS
Security and Presentation	• Global page: N/A • ATS page: Total on market volume share all securities traded on selected local venue (one line per selected venue) • Instruments page: On market volume share per venue of selected equity securities (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not Required
Metric Category	Post Trade
Algorithm	**For each trading day** 1.0 For each security, calculate and store the daily on market volume share for each of the six venues as below: $$SecVolumeShare_{v,s,d} = \frac{SecVolume_{v,s,d}}{\sum_1^V SecVolume_{v,s,d}}$$ 2.0 For each venue, calculate and store the daily on market volume share as below: 2.1 Aggregate daily on market volume across all qualified securities $$VenueVolume_{v,d} = \sum_1^S SecVolume_{v,s,d}$$ 2.2 Calculate the share for each venue as $$VenueVolumeShare_{v,d} = \frac{VenueVolume_{v,d}}{\sum_1^V VenueVolume_{v,d}}$$ 3.0 The following fields will be stored in the database at security level: 1) Date 2) Security 3) Venue 4) $SecVolume_{v,s,d}$ 5) $SecVolumeShare_{v,s,d}$ 4.0 The following fields will be stored in the database at venue level: 1) Date 2) Venue 3) $VenueVolume_{v,d}$ 4) $VenueVolumeShare_{v,d}$
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *v* stands for local ATS venue or national exchange • *d* stands for trading day 2) "All equity securities" refers all equity securities on primary venue 3) "Cross traded securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 2: Daily Percentage of NBBO Time by Venue

Event types	On-market trade
Data Source	TRTH Quotes and Trades database for local ATS
Security and Presentation	• Global page: N/A • ATS page: Daily percentage of NBBO time across all securities of selected local venue (one line per selected venue) • Instruments page: : Daily percentage of NBBO of selected equity securities (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not Required
Metric Category	Pre-trade
Algorithm	**On every quote update from any ATS venues** 1.0 Determine the national best bid/ask price for that security 1.1 Set the national best bid price (***$NBB_{s,d}$***) to be the highest best bid price among all ATS venues. 1.2 Set the national best ask price (***$NBA_{s,d}$***) to be the lowest best ask price among all ATS venues. 2.0 For each venue, compare its best bid/ask price to the prevailing ***$NBB_{s,d}$*** / ***$NBA_{s,d}$*** for that security. **For the bid side** 2.1 Check if the current venue has the starting time at national best bid price (***$NBBTimestamp_{v,s,d}$***) defined for that security. If yes then 2.1.1 Increase the venue's duration at national best bid price (***$NBBDuration_{v,s,d}$***) by the difference between the current time and ***$NBBTimestamp_{v,s,d}$***. 2.1.2 Reinitialize ***$NBBTimestamp_{v,s,d}$*** (so it is "undefined"). 2.2 If the current venue's best bid price equals ***$NBB_{s,d}$*** then 2.2.1 Set ***$NBBTimestamp_{v,s,d}$*** to the current time 2.2.2 Update the starting time at ***$NBB_{s,d}$*** for that venue (***$NBBTimestamp_{v,s,d}$***) to the current time **For the ask side** 1.1 Check if the current venue has the starting time at national best ask price (***$NBATimestamp_{v,s,d}$***) defined for that security. If yes then 1.1.1 Increase the venue's duration at national best bid/ask price (***$NBADuration_{v,s,d}$***) by the difference between the current time and ***$NBATimestamp_{v,s,d}$***. 1.1.2 Set***$NBATimestamp_{v,s,d}$*** to "undefined". 1.2 If the current venue's best bid price equals ***$NBA_{s,d}$*** then 1.2.1 Set ***$NBATimestamp_{v,s,d}$*** to the current time 1.2.2 Update the starting time at ***$NBA_{s,d}$*** for that venue (***$NBATimestamp_{v,s,d}$***) to the current time **At the end of the trading day** 2.0 Loop over all securities in each venue and 2.1 Calculate the percentage at total duration at national best bid/ask price for each security-venue pair as $$SecNBBADuration\%_{v,s,d} = \frac{NBBDuration_{v,s,d} + NBADuration_{v,s,d}}{(Close\ Time_{v,s,d} - Open\ Time_{v,s,d}) * 2}$$ 2.2 Calculate the daily total duration at national best bid/ask price and daily total trading hours across common securities for each venue as: $$VenueNBBDuration_{v,d} = \sum_{s=1}^{S} NBBDuration_{v,s,d}$$

MQ Dashboard Metrics

	$$VenueNBADuration_{v,d} = \sum_{s=1}^{S} NBADuration_{v,s,d}$$ $$VenueTotalHours_{v,d} = \sum_{s=1}^{S} Close\ Time_{v,s,d} - Open\ Time_{v,s,d}$$ 2.3 Calculate the percentage of total duration of national best bid and ask price for each venue across common securities as: $$VenueNBBADuration\%_{v,d} = \frac{VenueNBBDuration_{v,d} + VenueNBADuration_{v,d}}{2 * \sum_{v=1}^{N} VenueTotalHours_{v,d}}$$ 3.0 Store the following data fields at security level • Date • Security • ***NBBDuration$_{v,s,d}$*** for each venue • ***NBADuration$_{v,s,d}$*** for each venue • ***SecNBBADuration%$_{v,s,d}$*** for each venue 4.0 Store the following data fields in at venue level a) Date b) ***VenueNBBDuration$_{v,d}$*** for each venue c) ***VenueNBADuration$_{v,d}$*** for each venue d) ***VenueNBBADuration%$_{v,d}$*** for each venue
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *d* stands for trading day • *v* stands for local ATS venue or national exchange 2) "All equity securities" refers all equity securities on primary venue 3) "Cross equity securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 3: Order to Trade Ratio

Event types	On-market trade, on-market order
Data Source	TRTH Quotes and Trades database for local ATS and global markets TRTH Market Depth Database (10 levels) for local ATS and global markets
Security and Presentation	• Global page: Ratio of total number of orders to total number of trades of all securities traded on selected international exchange (one line per selected international exchange) • ATS page: Ratio of total number of orders to total number of trades of all securities from selected local venue (one line per selected local venue) • Instruments page: Ratio of total number of orders to total number of trades per venue of selected instrument (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Required. A decile version of the metric will be created separately (i.e., Order to Trade Ratio by Turnover Decile)
Metric Category	Pre-trade
Algorithm	**On every quote update (from the Market Depth Database)** 1.0 Check if there has been a trade since a second ago (trade data is from the Quotes and Trades Database). If no then assign a quote movement flag for each of the 10 price levels based on the following rules: 1.1 For each price level that exists in both the previous quote and the current quote, assign an up (down) flag if the volume for that price level increased (decreased); 1.2 For each price level that existed in the previous quote but not the new quote, assign a down flag 1.3 For each price level that exists in the new quote but not in the previous quote, assign an up flag 2.0 Decide whether the current quote was caused by an order entry, cancellation or amendment based on the following rules: 2.1 If this quote has one or more up flags and no down flags, increase the number of order entry (***EnterCount***$_{v,s,d}$) by one; 2.2 If this quote has one or more down flags and no up flags, increase the number of order cancellation (***CancelCount***$_{v,s,d}$) by one; 2.3 For all other cases, increase the number of order amendment (***AmendCount***$_{v,s,d}$) by one; **At the end of each day**, 3.0 For each security and exchange/venue pair, aggregate the order transaction counts (across the three order transaction types) and then calculate and store the order transactions to trade ratio. Securities that are note traded are excluded for the calculation of market level metric. $$SecOtoTRatio_{v,s,d} = \frac{EnterCount_{v,s,d} + AmendCount_{v,s,d} + CancelCount_{v,s,d}}{TradeCount_{v,s,d}}$$ 4.0 Finally calculate and store daily order transactions to trade ratio for each venue by the following formula. $$VenueOtoTRatio_{v,d} = \frac{\sum_{s=1}^{N} EnterCount_{v,s,d} + \sum_{s=1}^{N} AmendCount_{v,s,d} + \sum_{s=1}^{N} CancelCount_{v,s,d}}{\sum_{s=1}^{N} TradeCount_{v,s,d}}$$
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *d* stands for trading day • *v* stands for local ATS venue or national exchange • *n* stands for number of national best bid/ask price 2) "All equity securities" refers all equity securities on primary venue 3) "Cross equity securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 4 & 18: Effective Spread (bps) and Realised Spread (bps)

Event types	On-market trade
Data Source	TRTH Quotes and Trades database for global markets
Security and Presentation	• Global page: Value-weighted effective spread and realised spread of all securities traded on selected international exchange (one line per selected international exchange) • ATS page: N/A • Instruments page: Value-weighted effective spread and realised per local venue of selected instrument (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Required. A decile version of the metric will be created separately (i.e., Effective Spread (bps) by Turnover Decile and Realised Spread (bps) by Turnover Decile)
Metric Category	Pre-trade
Algorithm	**On every trade** 1.0 Calculate the midpoint price ($Mid_{v,s,d,t}$) immediately prior the current trade execution and store it into the midpoint price array for that security $Mid_{v,s,d,t} = (Best\ Bid\ Price_{v,s,d,t} + Best\ Ask\ Price_{v,s,d,t}) / 2$ 2.0 Store the following information into arrays for the current security • Store the value of the current trade ($Val_{v,s,d,t}$) • Store the direction of the current trade ($D_{v,s,d,t}$) ➢ Set $D_{v,s,d,t}$ to be 1 for buyer-initiated trade ➢ Set $D_{v,s,d,t}$ to be -1 for seller-initiated trade • Store the current trade price ($P_{v,s,d,t}$) **On every quote update** 3.0 Store the quote update time and the prevalent midpoint price ($Mid_{v,s,d,t}$) for that security **At the end of the day** 4.0 Loop over each security that has been traded on the current trading day 4.1 Calculate the value-weighted daily average relative effective spread ($EffSpr_{v,s,d}$) for each security 4.1.1 Calculate the daily turnover ($Turnover_{v,s,d}$) by aggregating the value of each trade for that security stored in 2.0 4.1.2 Calculate the value weight ($W_{v,s,d,t}$) of each trade for that security as $W_{v,s,d,t} = Val_{v,s,d,t} / Turnover_{v,s,d}$ 4.1.3 Finally loop over all trades for that security on the current trading day and calculate $EffSpr_{v,s,d}$ as $$EffSpr_{v,s,d} = \sum 200 * D_{v,s,d,t} * \frac{P_{v,s,d,t} - Mid_{v,s,d,t}}{Mid_{v,s,d,t}} * W_{v,s,d,t}$$ 4.2 Calculate and store the value-weighted daily average relative realised spread ($RealSpr_{v,s,d}$) for each security 4.2.1 Loop over all trades for that security on the current trading day and for each trade find the first available midpoint price GRACE_PERIOD minutes (i.e., $V_{i,t+10mins}$) after that trade 4.2.2 Calculate $RealSpr_{v,s,d}$ as $$RealSpr_{v,s,d} = \sum 200 * D_{v,s,d,t} * \frac{P_{v,s,d,t} - Mid_{v,s,d,t+grace\_period}}{Mid_{v,s,d,t}} * W_{v,s,d,t}$$ 5.0 Calculate the value-weighted daily average relative effective spread and realised spread for each

	venue/global exchange. $$VenueEffSpr_{v,d} = \frac{\sum_{s=1}^{S} EffSpr_{v,s,d} * Turnover_{v,s,d}}{\sum_{s=1}^{S} Turnover_{v,s,d}}$$ $$VenueRealSpr_{v,d} = \frac{\sum_{s=1}^{S} RealSpr_{v,s,d} * Turnover_{v,s,d}}{\sum_{s=1}^{S} Turnover_{v,s,d}}$$
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *d* stands for trading day • *v* stands for local ATS venue or national exchange • *n* stands for number of quote update 2) "All equity securities" refers all equity securities on primary venue 3) "Common equity securities" refers to common equity securities traded on all ATS venues 4) grace_period stands for the time period between the prevailing trade and the subsequent quote chosen for comparison. It's normally set at 10 minutes level.

MQ Dashboard Metrics

Metric 5: Intraday Volatility 1

Event types	Quotes
Data Source	TRTH Quotes and Trades database for global markets
Security and Presentation	• Global page: Value-weighted intraday volatility of all securities traded on selected international exchange (one line per selected international exchange) • ATS page: N/A • Instruments page: Value-weighted intraday volatility per local venue of selected instrument (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not required
Metric Category	Market Quality
Algorithm	**Every minute for each security** 1.0 Calculate midpoint price ($Mid_{v,s,d,t}$) $MidP_{v,s,d,t} = (Best\ Bid\ Price_{v,s,d,t} + Best\ Ask\ Price_{v,s,d,t}) / 2$ 2.0 If this is not the first midpoint price of the day stored for that security, calculate 1 minute midpoint log return ($MidPRet_{v,s,d,t}$) and add it to a distribution ($MidPRetDist_{v,s,d}$) $$MidPRet_{v,s,d,t} = \ln(\frac{MidP_{v,s,d,t}}{MidP_{v,s,d,t-1}})$$ **At the end of the day** 3.0 Loop over each security that has quotes during the day and perform the following. 3.1 Calculate and store the volatility (i.e., standard deviation) of the $MidPRetDist_{v,s,d}$ distribution for each security $$VolaMidPRet_{v,s,d} = Stdev(MidPRet_{v,s,d})$$ 3.2 Calculate the value weight ($W_{v,s,d,t}$) for each security as $W_{v,s,d} = SecTurnover_{v,s,d} / MktTurnover_{v,s,d}$ 3.3 Finally calculate and store the daily value-weighted 1-minutes midpoint return volatility as $$MktVolaMidPRet_{v,d} = \sum_{s=1}^{N} W_{v,s,d} * VolaMidPRet_{v,s,d}$$
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *d* stands for trading day • *v* stands for local ATS venue or national exchange • *n* stands for number of national best bid/ask price 2) "All equity securities" refers all equity securities on primary venue 3) "Common equity securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 6: Info Leakage

Event types	Trades and Quotes
Data Source	TRTH Quotes and Trades database for global markets
Security and Presentation	• Global page: Sum of info leakage cases' dollar value across all securities traded on selected international exchange (one line per selected international exchange) • ATS page: N/A • Instruments page: Sum of info leakage cases' dollar value per local venue of selected instrument (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not required
Metric Category	Market Quality
Algorithm	**On every information announcement** **1.0 Rule 1** If this announcement is the first announcement of the current trading day then 1.1 Check if the date of announcement falls within the date of the previous announcement + EVENT_WINDOW_AFTER_INFO trading days 1.1.1 If yes, remove the previous announcement from list of clean event window. 1.2 Check if the date of current announcement - EVENT_WINDOW_BEFORE_INFO trading days is after the date of the previous announcement. 1.2.1 If yes, store current announcement as a clean event. **At the end of the Alerting Period** **2.0 Loop over each information announcement in the list of clean events:** 2.1 When information announcement is released during the continuous trading period, set the check point of reference price (TRP) to be the release time of the information announcement: • RP_{t-n} represents the mid-point price of that security at TRP on trading day ***t-n*** from the pre-event window (i.e., from [t-6, t-0]); • RP_{t+n} represents the mid-point price of that security at TRP on trading day ***t+n*** from the post-event window (i.e., from [t-0, t+2]); 2.2 When information announcement is released before or after the continuous trading period, set the check point of reference price (TRP) to be the market close time before the announcement: • RP_{t-n} represent the close price of that security on trading day ***t-n*** from the pre-event window; • RP_{t+n} represent the close price of that security on trading day ***t+n*** from the post-event window; 2.3 Calculate **Alpha** and **Beta** by using historical trading data for the benchmarking period (please refer to the Benchmark section for details). **3.0 Pre-Event Window Analysis** 3.1 Calculate the daily log abnormal return of that security (ABR_{t-n}) for each trading day of the pre-event window where: $ABR_{t-n} = ln(RP_{t-n} / RP_{t-n-1}) - Alpha - (Beta*Market\ Return_{t-n})$ • If information announcement released after the continuous trading period, the date range of pre-event window is shifted right by one trading day (i.e., Pre-event window becomes t-5 to t+1) 3.2 Compare the daily ABR_{t-n} with the **Positive Abnormal Return Threshold** (cutoff_return_high) and **Negative Abnormal Return Threshold** (cutoff_return_low) and store the first ABR_{t-n} which exceeds either of two thresholds by looking back from t-0 and determine the direction of this daily abnormal return (See section 4.0 for reference),

3.2.1 **Rule 2** **IF**

- **No** daily ABR_{t-n} exceeds both thresholds

Then

The direction of daily abnormal return is deemed **flat** (i.e., "-");

3.2.2 **Rule 3** **IF**

- Daily ABR_{t-n} which is greater than Positive Abnormal Return Threshold; **AND**
- Daily ABR_{t-n} which is less than Negative Abnormal Return Threshold,

Then

The direction of daily abnormal return is deemed **inconsistent** (i.e., "?");

3.2.3 **Rule 4** **IF**

- Daily ABR_{t-n} is greater than Positive Abnormal Return Threshold; **AND**
- **No** daily ABR_{t-n} which is less than Negative Abnormal Return Threshold,

Then

The direction of daily abnormal return is deemed **up** (i.e., "/");

3.2.4 **Rule 5** **IF**

- **No** daily ABR_{t-n} which is greater than Positive Abnormal Return Threshold; **AND**
- **No** daily ABR_{t-n} which is less than Negative Abnormal Return Threshold,

Then

The direction of daily abnormal return is deemed **down** (i.e., "\");

3.3 Finally, all ABR_{t-n} of the pre-event window is summed and is stored as the **CAR** from t-6 to t-0 (See section 4.0 for reference).

4.0 Post-Event Window Analysis

4.1 Calculate the daily log abnormal return of that security (ABR_{t+n}) for each trading day from the post-event window as

$$ABR_{t-n} = ln(RP_{t-n} / RP_{t-n-1}) - Alpha - (Beta * Market\ Return_{t-n})$$

- If information announcement released after the continuous trading period, the date range of pre-event window is shifted right by one trading day (i.e., Pre-event window becomes t+1 to t+3

4.2 Compare the daily ABR_{t-n} with the **Positive Abnormal Return Threshold** (cutoff_return_high) and **Negative Abnormal Return Threshold** (cutoff_return_low) and store the first ABR_{t-n} which exceeds either of two thresholds by looking forward from t-0 and determine the direction of this daily abnormal return (See section 4.0 for reference),

4.2.1 **Rule 6** **IF**

- **No** daily ABR_{t-n} exceeds both thresholds

Then

The direction of daily abnormal return is deemed **flat** (i.e., "-");

4.2.2 **Rule 7** **IF**

- Daily ABR_{t-n} which is greater than Positive Abnormal Return Threshold; **AND**
- Daily ABR_{t-n} which is less than Negative Abnormal Return Threshold;

Then

The direction of daily abnormal return is deemed **inconsistent** (i.e., "?");

4.2.3 **Rule 8** **IF**

- Daily ABR_{t-n} is greater than Positive Abnormal Return Threshold; **AND**
- **No** daily ABR_{t-n} which is less than Negative Abnormal Return Threshold,

	Then The direction of daily abnormal return is deemed **up** (i.e., "/"); 4.2.4 **Rule 9 IF** • **No** daily ABR_{t-n} which is greater than Positive Abnormal Return Threshold; **AND** • **No** daily ABR_{t-n} which is less than Negative Abnormal Return Threshold, **Then** The direction of daily abnormal return is deemed **down** (i.e., "\"); 4.3 Finally, all ABR_{t-n} of the pre-event window is summed and is stored as the **CAR** from t-0 to t+2 (See section 4.0 for reference). **5.0 Determination of Price Sensitive Announcements** 5.1 **Rule 10 IF** • The total CAR across the pre- and post-event window (i.e., [t-6, t+2], is equal to or exceeds either **Positive CAR Threshold** or **Negative CAR Threshold** (See section 4.0 for reference), **Then** • The announcement will be recognised as **Price Sensitive Announcement**; **AND** • Store this information announcement and direction of daily abnormal return in both pre-event and post-event window. **6.0 Detection of Suspected Insider Trading Case** 6.1 Loop over the Price Sensitive Announcement stored for each security on the current trading day (t-n/t+n) 6.1.1 Set INCLUDE_INCONSISTANT_MOVES to be false so that cases with inconsistent price movement directions in the pre-event and post-event window will be excluded; 6.1.2 If all cases where the direction of abnormal return in both pre- and post-event window are in a same direction, Then **Trigger an info leakage alert;** *Otherwise* 6.1.3 If all cases where there is a direction of abnormal return in pre-event window is in the same direction as total CAR, which is summed up CAR both in pre- and post-event window, Then **Trigger an info leakage alert.**
Benchmark	**Principles for Threshold Determination** (1) The benchmarking period will be the MAX_SAMPLE_SIZE (currently set as 250 trading days) looking back from t-10 (BENCHMARK_START_OFFSET) to t -260; (2) If the number of trading days a security is traded during the benchmarking period is less than MIN_SAMPLE_SIZE (currently set as 150 trading days), the threshold will be undefined and no alerts will be ; (3) The top 1% observations from both tail will be dropped from the distribution of abnormal return over the benchmarking period in order to exclude outliers; (4) The number of standard deviations from the mean to use to calculate the threshold is determined by the ST_DEV user parameter.

	Rolling Benchmarking **1.0 For each trading day from the benchmarking period** 1.1 Calculate the daily return ($R_{i,d}$) for each security. $R_{i,d} = ln(Close\ Price_{i,d} / Close\ Price_{i,d-1})$ 1.2 Calculate the daily market return $RM_{i,d} = ln(Close\ Value\ of\ the\ Main\ Index_{d} / Close\ Value\ of\ the\ Main\ Index_{i,d-1})$ 1.3 Conduct a market model regression analysis on security's daily return over the daily market return. • $R_{i,d} = Alpha + Beta * Market\ Return_{d} + error\ term_{d}$ • "$error\ term_{d}$" represents the abnormal return of a security ($ABR_{i,d}$) on trading day d; 1.4 Store each $ABR_{i,d}$ into the historical daily abnormal return distribution for that security; 1.5 Randomly choose 1000 samples of nine trading days from the benchmarking period and calculate the cumulative abnormal return ($CAR_{i,n}$, n ranges from 1 to 1000) for each period by aggregating the $ABR_{i,d}$ for each day from the sample period and store $CAR_{i,n}$ into the historical bootstrapped CAR distribution for that security; **2.0 At the end of the benchmarking** 2.1 For each security, calculate the distribution mean and standard deviation for the historical bootstrapped CAR distribution. Then calculate the positive/negative CAR threshold for that security which will be used to identify price sensitive announcement in the alerting section. • Positive CAR Threshold = distribution mean + ST_DEV * standard deviation • Negative CAR Threshold = distribution mean – ST_DEV * standard deviation 2.2 For each security, drop off the top 1% observations from both tail of the historical daily abnormal return distribution and use the remaining observations to recreate the historical daily abnormal return distribution. Calculate the distribution mean and standard deviation for the recreated distribution. And then calculate the positive/negative abnormal return threshold as • Positive Abnormal Return Threshold = distribution mean + ST_DEV * standard deviation • Negative Abnormal Return Threshold = distribution mean - ST_DEV * standard deviation
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *d* stands for trading day • *v* stands for local ATS venue or national exchange • *n* stands for number of national best bid/ask price 2) "All equity securities" refers all equity securities on primary venue 3) "Common equity securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 7: On Market Value by Venue

Event types	On-market trade
Data Source	TRTH Quotes and Trades database for local ATS
Security and Presentation	• Global page: N/A • ATS page: Total on market value share of all securities of selected local venue (one line per selected venue) • Instruments page: On market value share per venue of selected equity securities (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not Required
Metric Category	Post Trade
Algorithm	**For each trading day** 1.0 For each security, calculate and store the daily on market value share for each of the six venues as below: $$SecValueShare_{v,s,d} = \frac{SecValue_{v,s,d}}{\sum_1^V SecValue_{v,s,d}}$$ 2.0 For each venue, calculate and store the daily on market value share as below: 2.1 Aggregate daily on market value across all qualified securities $$VenueValue_{v,d} = \sum_1^S SecValue_{v,s,d}$$ 2.2 Calculate the share for each venue as $$VenueValueShare_{v,d} = \frac{VenueValue_{v,d}}{\sum_1^V VenueValue_{v,d}}$$ 3.0 The following fields will be stored in the database at security level: 6) Date 7) Security 8) Venue 9) $SecValue_{v,s,d}$ 10) $SecValueShare_{v,s,d}$ 4.0 The following fields will be stored in the database at venue level: 5) Date 6) Venue 7) $VenueValue_{v,d}$ 8) $VenueValueShare_{v,d}$
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *v* stands for local ATS venue or national exchange • *d* stands for trading day 2) "All equity securities" refers all equity securities on primary venue 3) "Cross traded securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 8: On Market Trade Count by Venue

Event types	On-market trade
Data Source	TRTH Quotes and Trades database for local ATS
Security and Presentation	• Global page: N/A • ATS page: Total on market trade count share of all securities of selected local venue (one line per selected venue) • Instruments page: On market trade count share per venue of selected equity securities (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not Required
Metric Category	Post Trade
Algorithm	**For each trading day** 1.0 For each security, calculate and store the daily on market trade count share for each of the six venues as below: $$SecTradeCountShare_{v,s,d} = \frac{SecTradeCount_{v,s,d}}{\sum_1^V SecTradeCount_{v,s,d}}$$ 2.0 For each venue, calculate and store the daily on market trade count share as below: 2.1 Aggregate daily on market trade count across all qualified securities $$VenueTradeCount_{v,d} = \sum_1^S SecTradeCount_{v,s,d}$$ 2.2 Calculate the share for each venue as $$VenueTradeCountShare_{v,d} = \frac{VenueTradeCount_{v,d}}{\sum_1^V VenueTradeCount_{v,d}}$$ 3.0 The following fields will be stored in the database at security level: 1) Date 2) Security 3) Venue 4) $SecTradeCount_{v,s,d}$ 5) $SecTradeCountShare_{v,s,d}$ 4.0 The following fields will be stored in the database at venue level: 1) Date 2) Venue 3) $VenueTradeCount_{v,d}$ 4) $VenueTradeCountShare_{v,d}$
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *v* stands for local ATS venue or national exchange • *d* stands for trading day 2) "All equity securities" refers all equity securities on primary venue 3) "Cross traded securities" refers to common equity securities traded on all ATS venues

Metric 9: On Market Square-root Value by Venue

Event types	On-market trade
Data Source	TRTH Quotes and Trades database for local ATS
Security and Presentation	• Global page: N/A • ATS page: Total on market square-root value share of all securities of selected local venue (one line per selected venue) • Instruments page: On market square-root value share per venue of selected equity securities (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not Required
Metric Category	Post Trade
Algorithm	**On each trade** 1.0 Calculate and store the square-root trade value: $SqrtValue_{v,s,d,t} = \sqrt{Value_{v,s,d,t}}$ **At the end of the day** 2.0 Calculate the daily total on-market square-root value across cross traded securities for each venue () $VenueSqrtValue_{v,d} = \sum_{s=1}^{x}\sum_{t=1}^{n} SqrtValue_{v,s,d,t}$ 3.0 For each security, calculate and store the daily on market square-root value share for each local ATS venue (***SqrtValueShare***$_{v,s,d}$). $SqrtValueShare_{v,d} = \frac{\sum_{t=1}^{n} SqrtValue_{v,s,d,t}}{\sum_{s=1}^{x}\sum_{t=1}^{n} SqrtValue_{v,s,d,t}}$ 4.0 Finally calculate and store the daily on market square-root value share for each venue (***VenueSqrtValueShare***$_{v,s,d}$) $VenueSqrtValueShare_{v,d} = \frac{VenueSqrtValue_{v,d}}{\sum_{v=1}^{m} VenueSqrtValue_{v,d}}$
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *v* stands for local ATS venue or national exchange • *d* stands for trading day 2) "All equity securities" refers all equity securities on primary venue 3) "Cross traded securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 10: Scope by Venue

Event types	On-market trade
Data Source	TRTH Quotes and Trades database for local ATS
Security and Presentation	• Global page: N/A • ATS page: Total number of all securities of selected local venue (one line per selected venue) • Instruments page: N/A
Metric Interval	Daily
Per Decile	Not Required
Metric Category	Post Trade
Algorithm	**At the end of the trading day** 1.0 For each venue, count and store the number of securities that have traded during the day (***$ActiveSecCount_{v,d}$***). 2.0 Calculate and store the scope of trading for each venue (***$Scope_{v,d}$***). $$Scope_{v,d} = \frac{ActiveSecCount_{v,d}}{Max(ActiveSecCount_{v[1,n],d})}$$
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *v* stands for local ATS venue or national exchange • *d* stands for trading day 2) "All equity securities" refers all equity securities on primary venue 3) "Cross traded securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 11: Daily Percentage of Tightest Spread Time by Venue

Event types	On-market trade
Data Source	TRTH Quotes and Trades database for local ATS
Security and Presentation	• Global page: N/A • ATS page: Daily percentage of national spread time across all securities of selected local venue (one line per selected venue) • Instruments page: Daily percentage of national spread time of selected equity securities (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not Required
Metric Category	Pre-trade
Algorithm	**On every quote update from any ATS venues** 1.0 Determine the national best bid/ask price for that security 1.1 Set the national best bid price (***$NBB_{s,d}$***) to be the highest best bid price among all ATS venues. 1.2 Set the national best ask price (***$NBA_{s,d}$***) to be the lowest best ask price among all ATS venues. 2.0 For each venue, compare its best bid and best ask price to the prevailing ***$NBB_{s,d}$*** and ***$NBA_{s,d}$*** for that security. 2.1 Check if the current venue has the starting time at national spread (***$NSprTimestamp_{v,s,d}$***) defined for that security. If yes then 2.1.1 Increase the venue's duration at national spread (***$NSprDuration_{v,s,d}$***) by the difference between the current time and ***$NSprTimestamp_{v,s,d}$***. 2.1.2 Reinitialize ***$NSprTimestamp_{v,s,d}$*** (so it is "undefined"). 2.2 If the current venue's best bid price equals ***$NBB_{s,d}$*** and its best ask price equals ***$NBA_{s,d}$*** then 2.2.1 Set ***$NSprTimestamp_{v,s,d}$*** to the current time 2.2.2 Update the starting time at national spread for that venue (***$NSprTimestamp_{v,s,d}$***) to the current time **At the end of the trading day** 3.0 Loop over all securities in each venue and 3.1 Calculate the percentage at total duration at national spread for each security-venue pair as $$SecNSprDuration\%_{v,s,d} = \frac{NSprDuration_{v,s,d}}{(Close\ Time_{v,s,d} - Open\ Time_{v,s,d}) * 2}$$ 3.2 Calculate the daily total duration at national spread and total trading hours across all securities for each venue as: $$VenueNSprDuration_{v,d} = \sum_{s=1}^{S} NSprDuration_{v,s,d}$$ $$VenueTotalHours_{v,d} = \sum_{s=1}^{S} Close\ Time_{v,s,d} - Open\ Time_{v,s,d}$$ 3.3 Calculate the percentage of total duration of national best bid and ask price for each venue across common securities as: $$VenueNSprDuration\%_{v,d} = \frac{VenueNSprDuration_{v,d}}{\sum_{v=1}^{V} VenueTotalHours_{v,d}}$$ 4.0 Store the following data fields at security level • Date

	• Security • ***NSprDuration***$_{v,s,d}$ for each venue • ***SecNSprDuration%***$_{v,s,d}$ for each venue 5.0 Store the following data fields in at venue level • Date • ***VenueNSprDuration***$_{v,d}$ for each venue • ***VenueSprDuration%***$_{v,d}$ for each venue
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *d* stands for trading day • *v* stands for local ATS venue or national exchange 2) "All equity securities" refers all equity securities on primary venue 3) "Cross equity securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 12: Daily Percentage of NBBO Depth (dollar value) by Venue

Event types	On-market trade
Data Source	TRTH Quotes and Trades database for local ATS
Security and Presentation	• Global page: N/A • ATS page: Daily percentage of NBBO depth across all securities of selected local venue (one line per selected venue) • Instruments page: Daily percentage of NBBO depth of selected equity securities (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not Required
Metric Category	Pre-trade
Algorithm	**On every quote update from any ATS venues** 1.0 Determine the national best bid/ask price for that security 1.1 Set the national best bid price (***NBB$_{s,d}$***) to be the highest best bid price among all ATS venues. 1.2 Set the national best ask price (***NBA$_{s,d}$***) to be the lowest best ask price among all ATS venues. 2.0 For each venue, compare its best bid/ask price to the prevailing ***NBB$_{s,d}$***/***NBA$_{s,d}$*** for that security. **For the bid side** 2.1 Check if the current venue has the starting time at national best bid price (***NBBTimestamp$_{v,s,d}$***) defined for that security. If yes then 2.1.1 Increase the venue's dollar value depth at national best bid price (***NBBDepth$_{v,s,d}$***) by $NBBValue_{v,s,d} * (Current\ Time - NBBTimeStamp_{v,s,d})$ 2.1.2 Reinitialize ***NBBTimestamp$_{v,s,d}$*** and ***NBBValue$_{v,s,d}$*** (so they are "undefined"). 2.2 If the current venue's best bid price equals ***NBB$_{s,d}$*** then 2.2.1 Set ***NBBTimestamp$_{v,s,d}$*** to the current time 2.2.2 Update the starting time at ***NBB$_{s,d}$*** for that venue (***NBBTimestamp$_{v,s,d}$***) to the current time **For the ask side** 2.3 Check if the current venue has the starting time at national best bid price (***NBATimestamp$_{v,s,d}$***) defined for that security. If yes then 2.3.1 Increase the venue's dollar value depth at national best bid price (***NBADepth$_{v,s,d}$***) by $NBAValue_{v,s,d} * (Current\ Time - NBATimeStamp_{v,s,d})$ 2.3.2 Reinitialize ***NBATimestamp$_{v,s,d}$*** and ***NBAValue$_{v,s,d}$*** (so they are "undefined"). 2.4 If the current venue's best bid price equals ***NBA$_{s,d}$*** then 2.4.1 Set ***NBATimestamp$_{v,s,d}$*** to the current time 2.4.2 Update the starting time at ***NBA$_{s,d}$*** for that venue (***NBATimestamp$_{v,s,d}$***) to the current time

	At the end of the trading day 3.0 Loop over all securities in each venue and 3.1 Calculate the daily percentage of national best bid/ask depth for each security-venue pair as $$SecNBBADepth\%_{v,s,d} = \frac{NBBDepth_{v,s,d} + NBADepth_{v,s,d}}{\sum_{v=1}^{V} NBBDepth_{v,s,d} + \sum_{v=1}^{V} NBADepth_{v,s,d}}$$ 3.2 Calculate the daily time-weighted dollar value depth at national best bid/ask price across all securities for each venue as: $$VenueNBBDepth_{v,d} = \sum_{s=1}^{S} NBBDepth_{v,s,d}$$ $$VenueNBADepth_{v,d} = \sum_{s=1}^{S} NBADepth_{v,s,d}$$ 3.3 Calculate the daily percentage of total national best bid/ask depth for each venue across common securities as: $$VenueNBBADepth\%_{v,d} = \frac{VenueNBBDepth_{v,d} + VenueNBADepth_{v,d}}{\sum_{v=1}^{N} VenueNBBDepth_{v,d} + \sum_{v=1}^{N} VenueNBADepth_{v,d}}$$ 4.0 Store the following data fields at security level • Date • Security • ***NBBDepth$_{v,s,d}$*** for each venue • ***NBADepth$_{v,s,d}$*** for each venue • ***SecNBBADepth%$_{v,s,d}$*** for each venue 5.0 Store the following data fields in at venue level • Date • ***VenueNBBDepth$_{v,d}$*** for each venue • ***VenueNBADepth$_{v,d}$*** for each venue • ***VenueNBBADepth%$_{v,d}$*** for each venue
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *d* stands for trading day • *v* stands for local ATS venue or national exchange 2) "All equity securities" refers all equity securities on primary venue 3) "Cross equity securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 13: On Market Trade Size by Venue

Event types	On-market trade
Data Source	TRTH Quotes and Trades database for local ATS
Security and Presentation	• Global page: N/A • ATS page: Average on market trade size of all securities of selected local venue (one line per selected venue) • Instruments page: Average on market trade size per venue of selected equity securities (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not Required
Metric Category	Post Trade
Algorithm	**For each trading day** 1.0 For each security, calculate and store the daily average on market trade size or each of the six venues as below: $$SecTradeSize_{v,s,d} = \frac{SecTotalValue_{v,s,d}}{SecTradeCount_{v,s,d}}$$ 2.0 For each venue, calculate and store the daily on market value share as below: $$VenueTradeSize_{v,d} = \frac{\Sigma_1^S SecTotalValue_{v,s,d}}{\Sigma_1^S SecTradeCount_{v,s,d}}$$ 3.0 The following fields will be stored in the database at security level: 1) Date 2) Security 3) Venue 4) $SecTotalValue_{v,s,d}$ 5) $SecTradeCount_{v,s,d}$ 6) $SecTradeSize_{v,s,d}$ 4.0 The following fields will be stored in the database at venue level: 1) Date 2) Venue 3) $VenueTotalValue_{v,d}$ 4) $VenueTotalTradeCount_{v,d}$ 5) $VenueTradeSize_{v,d}$
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *v* stands for local ATS venue or national exchange • *d* stands for trading day 2) "All equity securities" refers all equity securities on primary venue 3) "Cross traded securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 14: Quoted Value near the BBA (within 50bps)

Event types	On-market trade 10-level quotes
Data Source	TRTH Quotes and Trades database for local ATS and global markets TRTH Market Depth Database (10 levels) for local ATS and global markets
Security and Presentation	• Global page: Value-weighted daily quoted value near BBO (within 50 bps) across all securities traded on selected international exchange (one line per selected international exchange) • ATS page: Value-weighted daily quoted value near BBO (within 50 bps) across all securities of selected local venue (one line per selected venue) • Instruments page: Time-weighted daily quoted value near BBO (within 50 bps) of selected equity securities (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not Required
Metric Category	Pre-trade
Algorithm	**On every quote update** 1.0 Determine the mid point price for that security in the current venue (***MidP$_{v,s,d}$***). If that security only has one level of quote and one side of the quote is missing, then skip step 1.0 and 2.0. $$MidP_{v,s,d} = \frac{LevelOneBid_{v,s,d} + LevelOneAsk_{v,s,d}}{2}$$ 2.0 Loop over 10-level quote messages of that security in the current venue and compare the bid/ask price at each level to the prevailing ***MidP$_{v,s,d}$*** for that security. 2.1 Check if the current venue has the midpoint starting time (***MidPTimestamp$_{v,s,d}$***) defined for that security. If yes then 2.1.1 Increase the total duration of mid point price (***TotalMidPDuration$_{v,s,d}$***) for that security in the current venue by $$Current\ Time - MidPTimeStamp_{v,s,d}$$ 2.1.2 Increase that security's quoted value near best bid (***QuotedValueNearBB$_{v,s,d}$***) by $$DepthNearBB_{v,s,d} * (Current\ Time - MidPTimeStamp_{v,s,d})$$ 2.1.3 Increase that security's quoted value near best ask (***QuotedValueNearBA$_{v,s,d}$***) by $$DepthNearBA_{v,s,d} * (Current\ Time - MidPTimeStamp_{v,s,d})$$ 2.1.4 2.1.5 Reinitialize ***DepthNearBB$_{v,s,d}$*** and ***DepthNearBA$_{v,s,d}$*** (so they are "undefined"). 2.1.6 Set ***MidPTimestamp$_{v,s,d}$*** to the current timestamp. **For the bid side** 2.2 Calculate the percentage difference between each quote level's bid price and the prevailing ***MidP$_{v,s,d}$***. The difference is calculated as $$\frac{MidP_{v,s,d} - BidAtLevel_{v,s,d,n}}{MidP_{v,s,d}} * 100\%$$ 2.2.1 If the difference is within 0.50%, increase the dollar value depth near the best bid price (***DepthNearBB$_{v,s,d}$***) by $$BidAtLevel_{v,s,d,n} * VolumeAtLevel_{v,s,d,n}$$

	For the ask side 2.3 Calculate the percentage difference between each quote level's ask price and the prevailing ***MidP$_{v,s,d}$***. The difference is calculated as $$\frac{AskAtLevel_{v,s,d,n} - MidP_{v,s,d}}{MidP_{v,s,d}} * 100\%$$ 2.3.1 If the difference is within 0.50%, increase the dollar value depth near the best ask price (***DepthNearBA$_{v,s,d}$***) by **At the end of the trading day** 3.0 Loop over all securities in each venue and 3.1 Calculate the time-weighted daily quoted value near BBA for each security-venue pair as $$SecQuotedValueNearBBA_{v,s,d} = \frac{QuotedValueNearBB_{v,s,d} + QuotedValueNearBA_{v,s,d}}{TotalMidPDuration_{v,s,d}}$$ 3.2 Calculate the value-weighted daily quoted value near BBA across all securities for each venue and global exchange as: $$VenueQuotedValueNearBBA_{v,d} = \frac{\sum_{s=1}^{S} SecQuotedValueNearBBA_{v,s,d} * SecValue_{v,s,d}}{\sum_{s=1}^{S} SecSecValue_{v,s,d}}$$ 4.0 Store the following data fields at security level • Date • Security • ***DepthNearBB$_{v,s,d}$*** for each venue/global exchange • ***DepthNearBA$_{v,s,d}$*** for each venue/global exchange • ***TotalMidPDuration$_{v,s,d}$*** for each venue/global exchange • ***SecQuotedValueNearBBA$_{v,s,d}$*** for each venue/global exchange 5.0 Store the following data fields in at venue/global exchange level • Date • ***VenueValueNearBBA$_{v,d}$*** for each venue/global exchange
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *d* stands for trading day • *v* stands for local ATS venue or national exchange • n stands for nth level of quote 2) "All equity securities" refers all equity securities on primary venue 3) "Cross equity securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 15: Number of Quote Updates

Event types	10-Level Quotes
Data Source	TRTH Quotes and Trades database for local ATS and global markets TRTH Market Depth Database (10 levels) for local ATS and global markets
Security and Presentation	• Global page: Number of 10-level quote updates across all securities traded on selected international exchange (one line per selected international exchange) • ATS page: Number of 10-level quote updates across all securities of selected local venue (one line per selected venue) • Instruments page: Number of 10-level quote updates of selected equity securities (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not Required
Metric Category	Pre-trade
Algorithm	**On every quote update** 1.0 Increase the number of 10-level quote update ($\boldsymbol{QuoteCount_{v,s,d}}$) by one for that security in the current venue. **At the end of the trading day** 2.0 Loop over all securities in each venue/global exchange and 2.0 Calculate the total number of 10-level quotes across all securities for each venue and global exchange as: $VenueQuoteCount_{v,d} = \sum_{s=1}^{S} QuoteCount_{v,s,d}$ 3.0 Store the following data fields at security level • Date • Security • $\boldsymbol{QuoteCount_{v,s,d}}$ for each venue/global exchange 4.0 Store the following data fields in at venue/global exchange level • Date • $\boldsymbol{VenueQuoteCount_{v,d}}$ for each venue/global exchange
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *d* stands for trading day • *v* stands for local ATS venue or national exchange 2) "All equity securities" refers all equity securities on primary venue 3) "Cross equity securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 16 & 17 - Quoted Spread ($) and Quoted Spread (bps)

Event types	On-market trade
Data Source	TRTH Quotes and Trades database for global markets
Security and Presentation	• Global page: Value-weighted quoted spread of all securities traded on selected international exchange (one line per selected international exchange) • ATS page: N/A • Instruments page: Time-weighted quoted spread per local venue of selected instrument (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Required. A decile version of the metric will be created separately (i.e., Quoted Spread ($) by Turnover Decile and Quoted Spread (bps) by Turnover Decile)
Metric Category	Pre-trade
Algorithm	**On every quote update** 1.0 Calculate the prevailing dollar value quoted spread (***$QSpr_{v,s,d,n}$***) by using the formula below and store it into the quoted spread array for that security. $$QSpr_{v,s,d,n} = AskPr_{v,s,d,n} - BidPr_{v,s,d,n}$$ 2.0 Calculate the prevailing relative quoted spread (***$RelQSpr_{v,s,d,t}$***) and store it into the relative quoted spread array for that security. The formula to use is $$RelQSpr_{v,s,d,n} = \frac{AskPr_{v,s,d,n} - BidPr_{v,s,d,n}}{MidPoint_{v,s,d,n}}$$ Where • ***Mid-point Price$_{v,s,d,n}$*** = (***Best Ask Price$_{v,s,d,n}$ +Best Bid Price$_{v,s,d,n}$***) / 2 3.0 Store the time of the current quote update (***$QTime_{v,s,d,n}$***) into the quote update time array for that security. **At the end of the day** 4.0 Loop over each security that has had quote updates on the current trading day 4.1 Calculate the time weight (***$TW_{v,s,d,n}$***) of each quote update on the current trading day for that security 4.1.1 For quote updates other than the last one on the current trading day, $$TW_{v,s,d,n} = \frac{QTime_{v,s,d,n+1} - QTime_{v,s,d,n}}{\sum_{n=1}^{N} QTime_{v,s,d,n+1} - QTime_{v,s,d,n}}$$ Note: For the last quote update on the current trading day, ***$QTime_{v,s,d,n+1}$*** = Market Close Time 4.2 Calculate time-weighted daily average dollar value relative quoted spread (***$TimeWeightedQSpr_{v,s,d,n}$***) for the that security-venue pair on the current trading day as $$TimeWeightedQSpr_{v,s,d,n} = \sum_{t=1}^{T} RelSpr_{v,s,d,n} * TW_{v,s,d,n}$$

	4.3 Calculate time-weighted daily average relative quoted spread ($TimeWeightedRelQSpr_{v,s,d,n}$) for the that security-venue pair on the current trading day as $$TimeWeightedRelQSpr_{v,s,d,n} = \sum_{n=1}^{N} RelQSpr_{v,s,d,n} * TW_{v,s,d,n}$$ 5.0 Finally calculate the value-weighted daily average dollar value quoted spread and relative spread across all securities for each venue/global exchange. $$VenueTimeWeightedQSpr_{v,d} = \frac{\sum_{s=1}^{S} TimeWeightedQSpr_{v,s,d} * Turnover_{v,s,d}}{\sum_{s=1}^{S} Turnover_{v,s,d}}$$ $$VenueTimeWeightedRelQSpr_{v,d} = \frac{\sum_{s=1}^{S} TimeWeightedRelQSpr_{v,s,d} * Turnover_{v,s,d}}{\sum_{s=1}^{S} Turnover_{v,s,d}}$$
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *d* stands for trading day • *v* stands for local ATS venue or national exchange • *n* stands for number of quote upate 2) "All equity securities" refers all equity securities on primary venue 3) "Common equity securities" refers to common equity securities traded on all ATS venues 4) grace_period stands for the time period between the prevailing trade and the subsequent quote chosen for comparison. It's normally set at 5, 10 or 15 minutes level.

MQ Dashboard Metrics

Metric 21: Intraday Volatility 2

Event types	Quotes
Data Source	TRTH Quotes and Trades database for global markets
Security and Presentation	• Global page: Value-weighted volatility (High vs Low) of all securities traded on selected international exchange (one line per selected international exchange) • ATS page: N/A • Instruments page: Value-weighted volatility (High vs Low)per local venue of selected instrument (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not required
Metric Category	Market Quality
Algorithm	**At the end of the day** 1.0 Loop over each security that has trades during the day and perform the following. 1.1 Calculate and store the volatility (i.e., High vs Low) for each security $$VolaHigh\_Low_{v,s,d} = \frac{HighPrice_{v,s,d} - LowPrice_{v,s,d}}{(HighPrice_{v,s,d} + LowPrice_{v,s,d})/2}$$ 1.2 Calculate the value weight ($W_{v,s,d,t}$) for each security as $$W_{v,s,d} = SecTurnover_{v,s,d} / VenueTurnover_{v,s,d}$$ 1.3 Finally calculate and store the daily value-weighted volatility as $$VenueVolaHigh\_Low_{v,d} = \sum_{s=1}^{N} W_{v,s,d} * VolaMidPRet_{v,s,d}$$
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *d* stands for trading day • *v* stands for local ATS venue or national exchange 2) "All equity securities" refers all equity securities on primary venue 3) "Common equity securities" refers to common equity securities traded on all ATS venues

Metric 22: Interday Close Price Volatility

Event types	Quotes
Data Source	TRTH Quotes and Trades database for global markets
Security and Presentation	• Global page: Standard deviation of log daily close price changes across of all securities traded on selected international exchange (one line per selected international exchange) • ATS page: N/A • Instruments page: N/A
Metric Interval	Daily
Per Decile	Not required
Metric Category	Market Quality
Algorithm	**At the end of the trading day** 1.0 For each security, calculate the log daily close price change and add it to a distribution (***ClosePrChgDist$_{v,d}$***). The log daily close price change is calculated as $$ClosePrChg_{v,s,d} = \ln \frac{ClosePrice_{v,s,d}}{ClosePrice_{v,s,d-1}}$$ 2.0 Calculate and store the standard deviation (volatility) of the log daily close price change distribution (***ClosePrChgDist$_{v,d}$***).
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *d* stands for trading day • *v* stands for local ATS venue or national exchange 2) "All equity securities" refers all equity securities on primary venue 3) "Common equity securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 23: Quote Stuffing

Event types	10-Level Quotes
Data Source	TRTH Market Depth Database (10 levels) for local ATS and global markets
Security and Presentation	• Global page: Number of quote stuffing alerts across all securities traded on selected international exchange (one line per selected international exchange) • ATS page: N/A • Instruments page: Number of quote stuffing alerts of selected equity securities (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not Required
Metric Category	Market Quality
Algorithm	**On every quote update** 1.0 Increase the number of quote updates (***MinuteQuoteCount$_{v,s,d,m}$***) for that security in the current minute interval by one **At the end of the trading day** 2.0 Loop over all minute intervals for each security in every venue/global exchange and 2.0 trigger a quote stuffing alert if ***MinuteQuoteCount$_{v,s,d,m}$*** for a minute interval exceeds the threshold (***QuoteStuffingThreshold$_{v,s,d}$***) and increase the number of quote stuffing case (***QuoteStuffingCount$_{v,s,d}$***) for that security on the current day by one 2.1 Store the following information for each alert • Venue/global exchange • Date • Security • Minute interval (e.g., the whole minute in HH:MM format) • ***MinuteQuoteCount$_{v,s,d,m}$*** • ***QuoteStuffingThreshold$_{v,s,d}$*** 2.2 Also store ***QuoteStuffingCount$_{v,s,d}$*** for each security at the end of the loop 3.0 For the venue/global exchange level, aggregate ***QuoteStuffingCount$_{v,s,d}$*** across all securities. $VenueQuoteStuffingCount_{v,d} = \sum_{s=1}^{N} QuoteStuffingCount_{v,s,d}$
Benchmark	**For the past 20 trading days** 1.0 Construct a 20-day distribution of number of quote updates (***HistQuoteCountDist$_{v,s,d}$***) per minute interval for each stock 2.0 Calculate the threshold for the quote stuffing alert to trigger for a stock as $QuoteStuffingThreshold_{v,s,d} = Mean(HistQuoteCountDist_{v,s,d}) + 20 * Stdev(HistQuoteCountDist_{v,s,d})$
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *m stands for the minute in which the quote update is submitted* • *d* stands for trading day • *v* stands for local ATS venue or national exchange 2) "All equity securities" refers all equity securities on primary venue 3) "Cross equity securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

MQ Dashboard Metrics

Metric 24: Dislocation of EOD Price

Event types	Trades and Quotes
Data Source	TRTH Quotes and Trades database for global markets
Security and Presentation	• Global page: Sum of dislocation of EOD price cases' dollar value across all securities traded on selected international exchange (one line per selected international exchange) • ATS page: N/A • Instruments page: Sum of dislocation of EOD price cases' dollar value per local venue of selected instrument (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Not required
Metric Category	Market Quality
Algorithm	**At the end of the continuous trading period** 1.0 Loop over all securities being traded on the current trading day and store the following information for each security: • Last trade price (RP_t) • Last available trade price at 15 minutes ($RP_{t-15mins}$) before the continuous trading period ends • The best bid price and the best ask price at the end of the continuous trading period 2.0 Assess the price movement during the last 15 minutes of the trading day 2.1 Rule 1 If RP_t is equal to or greater than the best ask price at the end of the continuous trading period, then store the difference between RP_t and that best ask price as ***Tick_Movement***; 2.2 Rule 2 If RP_t is equal to or less than the best bid price at the end of the continuous trading period, then store the difference between that best bid price and the RP_t as ***Tick_Movement***. 3.0 Determine the daily price change ($Pchg_t$) as: $Pchg_t = \ln(RP_t / RP_{t-15mins})$ 4.0 Capture suspected dislocation of EOD price cases occurred during the continuous trading period: 4.1 Rule 3 If $RP_{t-15mins}$ is defined (i.e., $RP_{t-15mins}$ is greater than \$0) **AND** 4.2 Rule 4 The positive/negative price change threshold for the continuous trading period is defined in the benchmarking section for that security **AND** 4.3 Rule 5 $PChg_t$ exceeds the positive/negative price change threshold for the continuous trading period **AND** 4.4 Rule 6 The ***Tick_Movement*** is defined for that security and its absolute value is greater than 0 (i.e., the last price of that security from the continuous trading period is outside the spread at the end of the continuous trading period); **THEN** 4.5 It will be classified as a suspected Dislocation of EOD price case occurred during the continuous trading period and stored in the suspected cases list I. **For securities exchanges that have closing auction, at the end of the closing auction** 5.0 Loop over all securities being traded on the current trading day and store followed information: • Close price ($RP_{auction}$) • Last trade price (RP_t) prior the first auction trade • Last available trade price at 15 minutes ($RP_{t-15mins}$) before the continuous trading period ends • The best bid price and the best ask price at the end of the continuous trading period 6.0 Filter out false positives stored in the suspected cases list of dislocation of EOD price occurred during the continuous trading period

	6.1 For each of the cases stored in suspected cases list I, recalculate ***Pchg%*** by using the auction as $Pchg_t = \ln(RP_{auction} / RP_{t-15mins})$ 6.2 Rule 7 If the $RP_{auction}$ is defined **AND** 6.3 Rule 8 The recalculated $Pchg_t$ in 6.1 does not exceed the positive/negative price change threshold defined in the benchmarking period **THEN** 6.4 It will be removed from the suspected cases list I. 7.0 Capture suspected dislocation of EOD price cases occurred during the closing auction: 7.1 Rule 10 If $RP_{auction}$ is equal to or greater than the best ask price at the end of the continuous trading period, then store the difference between $RP_{auction}$ and that best ask price as ***Tick_Movement***; 7.2 Rule 11 If $RP_{auction}$ is equal to or less than the bid at the end of the continuous trading period, then store the difference between that best bid price and the $RP_{auction}$ as ***Tick_Movement***. 7.3 Determine the daily price change ($Pchg_t$) as: $Pchg_t = ln(RP_{auction} / RP_t)$ 7.4 Rule 12 If RP_t and $RP_{auction}$ are defined (i.e., greater than \$0); 7.5 **AND** 7.6 Rule 13 If The positive/negative price change threshold for the closing auction is defined for that security in the benchmarking section **AND** 7.7 Rule 14 If $Pchg_t$ exceeds the positive/negative price change thresholds for the closing auction; **AND** 7.8 Rule 15 If the ***Tick_Movement*** is defined (i.e., it's absolute value is greater than 0) **AND** 7.9 Rule 16 If the total turnover in the Closing Auction Period of that security is greater than \$0, **THEN** 7.10 It will be classified as a suspected case occurred in closing auction and stored in the list II. **After the continuous trading period on *t+1* starts** 8.0 Loop over all securities that have suspected dislocation cases detected on t and store the following information for each of those securities • Opening price on ***t+1*** (RP_{t+1}) • Last trade price on t or last trade price before closing auction for markets that have closing auction (RP_t) • Close price ($RP_{auction}$) for market that have closing auction $RP_{auction}$ • Last available trade price at 15 minutes ($RP_{t-15mins}$) before the continuous trading period ends 9.0 Assess the price movement on t+1 and filter out false positives stored in the suspected cases list of dislocation of EOD price occurred during the continuous trading period and during the closing auction 9.1 For each security that has suspected cases occurred in the continuous trading period stored on ***t***, calculate $Pchg_{t+1}$ as $Pchg_{t+1} = (RP_t - RP_{t+1}) / (RP_t - RP_{t-15mins}) * 100\%$ 9.2 For each security that has suspected cases occurred in the closing auction stored on t , calculate $Pchg_{t+1}$ as $Pchg_{t+1} = (RP_{auction} - RP_{t+1}) / (RP_{auction} - RP_t) * 100\%$ 9.3 Rule 13 If $Pchg_{t+1}$ is equal to or greater than 50% **THEN** 9.4 Trigger an Dislocation of EOD Price Alert
Benchmark	**Principles for Threshold Determination**

	1.0 The count of the valid sample security should be greater than the MIN_SAMPLE_SIZE and less than MAX_SAMPLE_SIZE prior to the BENCHMARK_WINDOW 2.0 We analyse two observation periods: the continuous trading period and the close auction period, and respectively, there will be two sets of threshold. **Rolling Benchmark** **3.0 Calculate the daily price distribution and then store them into a merged distribution for the benchmarking period, prepared for the threshold** 3.1 Store followed daily information: • Last price in every PRIOR_PERIOD_LENGTH ahead of the current price during the continuous trading period ($S_{i,t-15mins}$); and • Current stock price ($S_{i,t}$); and 3.2 Calculate the price change by using Smarts Model : $Pchg\% = ln(S_{i,t} / S_{i,t-15mins})$ 3.3 Store those ***Pchg%*** into the Daily Price Distribution for each security; 3.4 Store the daily price distribution to the Merged Distribution. 4.0 **For markets that have closing auction, calculate the auction price distribution and then store them into an auction rolling distribution, prepared for the threshold** 4.1 Loop over all the securities and store close price of the day as the current price (***Close Price***$_t$) and last trade price before the closing auction starts (***LTP***$_t$); 4.2 Calculate the price change between current price and past price $Pchg\% = ln(Close\ Price_t / LTP_t)$ 4.3 Store those ***Pchg%*** into the Auction Price Distribution for each security; 4.4 Store the Auction Price Distribution to the Auction Rolling Distribution. **At End** **5.0 Calculate the price movement threshold for the continuous trading period:** 5.1 Calculate the positive and negative thresholds for the continuous trading period: 5.1.1 Rule 1 If there are over 25 Daily Price Distributions stored in the Merged Distribution, Then **Positive Threshold** = Merged Merged Distribution Mean * 100% + Standard Deviation * STDEV_CUT_OFF *100% **Negative Threshold** = Merged Distribution Mean * 100% - Standard Deviation * STDEV_CUT_OFF *100% **6.0 Calculate the price movement threshold for closing auction:** 6.1 Calculate the positive and negative thresholds for the closing auction period: 6.1.1 Rule 2 If there are over 25 Auction Price Distributions stored in the Auction Rolling Distribution, Then **Positive threshold** = Auction Rolling Distribution Mean * 100% + Standard Deviation * STDEV_CUT_OFF *100% **Negative threshold** = Auction Rolling Distribution Mean * 100% - Standard Deviation * STDEV_CUT_OFF *100%
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *t* stands for quote update time • *d* stands for trading day • *v* stands for local ATS venue or national exchange • *n* stands for number of national best bid/ask price

	2) "All equity securities" refers all equity securities on primary venue 3) "Common equity securities" refers to common equity securities traded on all ATS venues

MQ Dashboard Metrics

Metric 25: Market Resiliency

Event types	Trades and Quotes
Data Source	TRTH Quotes and Trades database for global markets
Metric Interval	Daily
Metric Category	Market Efficiency - Resilience
Algorithm	**At the beginning of continuous trading** 1.0 Store the midpoint for each security (MP_s) **Every 5 seconds after continuous trading commences** 2.0 Increase the number of 5-second intervals (n) by one 3.0 Store the following information for each active security: • Last trade price in interval n ($LP_{s,n}$) • When n is at or greater than 12 (i.e., **1** minute into continuous trading), calculate and store the signed order imbalance in the past minute ($OI_{s,n}$) for each active security. Signed order imbalance is defined in the Definitions section below. 4.0 When n is greater than 16 (i.e., **1** minute an 25 seconds into continuous trading), estimate and store the error term ($\varepsilon_{s,n}$) from the fifth-order auto-regression of order imbalance ($OI_{s,n}$) for each active security. The auto-regression model is specified below: $$OI_{S,n} = \alpha_{s,n} + \beta_{s,1} OI_{s,n-1} + \beta_{S,2} OI_{s,n-2} + \beta_{S,3} OI_{s,n-3} + \beta_{S,4} OI_{s,n-4} + \beta_{S,5} OI_{s,n-5} + \varepsilon_{s,n}$$ **At the end of the trading day** 5.0 Use the Generalised Method of Moment to estimate the permanent price impact (λ), the temporary price impact (γ) and the resiliency (θ) of the day for each active security. The model used is specified below and the number of lags used is 60 (i.e., 5 minutes back. In other words, the first observation is taken at the 73rd 5-second interval or 6 minutes and 5 seconds after continuous trading commences). $$LP_{s,n} - MP_s = \alpha_s + \gamma_s \sum_{j=n-k}^{n} \theta_s^{n-j} OI_{s,j} + \lambda_s \sum_{j=1}^{n} q_{s,j} + \varepsilon_{s,n}$$ **Where** • $LP_{s,n}$ is the last trade price in the nth 5-second period • MP_s is the mid-point of security S at the start of the continuous trading session • γ_s is the estimator for temporary price impact in security S • θ_s is the estimator for resiliency of security S • $OI_{s,j}$ is the signed order imbalance of security S for the jth 5-second interval • λ_s is the permanent price impact in security S • $q_{s,j}$ is the error term estimated from section 4.0 above for the jth 5-second interval 6.0 To estimate the resiliency for each security at the 5 minutes, 10 minutes and 15 minutes level, we take the power of (N minutes/5 seconds) to the estimator for resiliency (θ_s) of the day. For example, the resiliency measure for the 5 minutes level is θ_s^{60} 7.0 To estimate the market level resiliency of the day, we sum the turnover-weighted resiliency across all securities.
Definitions	1) The subscript of data arrays referred in this document are • s stands for security;

	2) Signed Order Imbalance ($OI_{s,n}$) is calculated in two steps. • Firstly, for the n^{th} 5-second interval, take difference between buyer-initiated and seller-initiated volume in the past 1 minute. • Secondly, normalise the singed volume difference by subtracting the mean and dividing by the standard deviation of order imbalance observed during the same minute from the past 30 trading days.

Metric 25. Price Discovery Metrics

Event types	On-market trades
Data Source	TRTH Quotes and Trades database for global markets
Security and Presentation	• Global page: N/A • ATS page: Price discovery metrics of all securities of selected local venue (one line per selected venue) • Instruments page: Price discovery metrics per venue of selected equity securities (number of lines per instrument equals the number of venues the instrument trades on)
Metric Interval	Daily
Per Decile	Yes
Metric Category	Market Efficiency - Transaction Cost
Concepts and Terminologies	**1.0 Price discovery has two fundamental aspects** 1.1 Relative speed refers how quickly a price series reflects new information about the fundamental value of the underlying asset. 1.2 Relative noise refers to bid-ask bounce (i.e., the price bounces back and forth between bid and ask), tick discreteness (i.e., the minimum price movement), and temporary deviations from intrinsic value due to illiquid market etc. **2.0 Test Arbitrage-free Cointegration[1] relationship** 2.1 **Arbitrage-free Cointegration** refers that prices in different venues for the same asset can deviate from one another in the short run due to trade frictions (such as taxes, bid-ask spread, brokerage fees, market impact costs etc.), but will converge in the long run to reflect the fundamental value of the same asset. 2.2 To test arbitrage-free cointegration, the **Johansen Maximum Likelihood Procedure[2]** will be conducted (for more details, please refer to http://en.wikipedia.org/wiki/Johansen_test). **3.0 Evolution of Price Discovery metrics** 3.1 Traditional metrics, such as the Information Share and Common Factor Share, cannot essentially distinguish between relative speed and relative noise in the process of price discovery. 3.2 Three recently-developed metrics, the Permanent Information Impounding, Price Discovery Efficiency and information Leadership Share, are derived from the traditional price discovery metrics but disentangle the impact of relative speed from relative noise.
Algorithm	**1.0 Collect trade tuples for the same security in multiple venues** 1.1 Conduct the **Johansen Maximum Likelihood Procedure** to test the existence of cointegration relationship among trading across multiple venues for the same security. Upon confirmation, calculate daily Information Share (IS) and Common Factor Share (CFS) metrics as below. 1.2 Using the **FILLFORWARD** sampling procedure to form trade tuples for ***IS*** 1.2.1 Concept 1.2.1.1 The frequency of trading varies from venue to venue as well as time to time. When lining-up trades from a less liquid venue against trades from the most liquid venue by timestamp to create trade tuple, the

[1] Please refer to Appendix 1 for more details on Cointegration

[2] In general, Johansen Maximum Likelihood procedure produces two statistics (i.e., maximum eigenvalue statistic and trace test statistic) to determine the existence of cointegration. For the purpose of price discovery analysis, maximum eigenvalue statistic is best for determining the exact number of cointergating vectors.

less liquid venue will have gaps in its queue of trades. Such gaps cause cross-correlation in the error terms derived from the vector error correction model (VECM) for any pair of less liquid and most liquid venue. FILLFORWARD is the procedure designed to mitigate this problem.

1.2.2 Procedure

1.2.2.1 Generate time grid for each venue based on the following rules.

- For every security, aggregate the number of trades across all trading venues each day;
- If the aggregated number of trade is one or above per minute on average for that security, a 10-second grid is used for each venue;
- If the aggregated number of trade is one or above every five minutes on average for that security, a 60-second grid is used for each venue.
- If there are fewer trades than that, the security is skipped across all trading venues.

1.2.2.2 Fill every grid in each venue with the latest available trade price.

- The filling process for each venue starts with the first grid with trade(s).
- If there are one or more trades in a grid, the price of last trade is used;
- If there are no trades in a grid, trade price from previous grid is used.

1.3 Using synchronous **MINSPAN** sampling procedure to form trade tuples for ***CFS***

1.3.1 Concept

1.3.1.1 The MINSPAN sampling procedure generates trade tuples looking forward and back around observations to find the minimum time span between trades on all trading venues and maximising the number of tuples.

1.3.2 Procedure

1.3.2.1 The MINSPAN procedure is only conducted for securities with at least one trade every five minutes on average in every venue.

1.3.2.2 The MINSPAN procedure is primarily based on the tie-breaking rules[3] described below.

- For venue V_i, aggregate trades from all other venues (V_i^c) and compare it to the trade count of V_i;
- Take the slower venue (the one with fewer trades) and for each of its trade (i.e., principal trade), find the nearest trade in the faster venue from both directions and use its price to form tuple with the principal trade;
- If the nearest trade from each direction are equally distant to the principal trade, the price of the nearest trade in the forward direction is used to form tuple with the principal trade.

2.0 Loop over all securities being traded in multiple venues on the current trading day and calculate price discovery metrics for each security and each venue[4]

2.1 Information Share ($IS_{v,s,d}$)

2.1.1 ***IS*** consists of contemporaneous response to both permanent and transitory shocks;

2.1.2 Typical reasons for high ***IS*** in a venue (V_i) are

[3] The tie-breaking rules come into play more often than might be expected, for two separate reasons. Firstly, when a market order is matched against multiple limit orders, there will be a sequence of transactions with the same time but different prices. Secondly, the time tick is finite (or at least recorded that way), so there will often be a difference which is a small multiple of that, resulting in the nearest trades in both directions being equally distant

[4] Please refer to Appendix 2 and 3 for more details

- price series in that venue impound greater permanent information ($\delta^{P}_{0,s,1}$); or
- Strong response to transitory shocks ($\delta^{T}_{0,s,2}$) in competing venue(s); or
- both ($\delta^{P}_{0,s,1}$ and $\delta^{T}_{0,s,2}$).

2.1.3 The formula for ***IS*** is

$$IS_{1,s,d} = \frac{\delta^{P}_{0,s,1}\delta^{T}_{0,s,2}}{\Delta} \quad , \quad IS_{1,s,d} = \frac{-\delta^{T}_{0,s,1}\delta^{P}_{0,s,2}}{\Delta} , \quad \Delta = \delta^{P}_{1,s,t}\delta^{T}_{2,s,t} - \delta^{T}_{1,s,t}\delta^{P}_{2,s,t}$$

2.2 Common Factor Share ($\mathbf{CFS_{v,s,d}}$)

2.2.1 ***CFS*** measures the proportion of price innovations in competing venue(s) that are driven by transitory shocks.

2.2.2 The formula for ***CFS*** is

$$CFS_{1,s,d} = \frac{\delta^{T}_{0,s,2}}{\Delta} \quad , \quad CFS_{2,s,d} = \frac{\delta^{T}_{0,s,1}}{\Delta} \quad ,$$

$$\Delta = \delta^{P}_{1,s,t}\delta^{T}_{2,s,t} - \delta^{T}_{1,s,t}\delta^{P}_{2,s,t}$$

2.3 Permanent Information Impounding ($PII_{v,s,d}$)

2.3.1 ***PII*** is the ratio of permanent shocks in venue V_i to the competing venue(s)

2.3.2 The formula for ***PII*** is

$$PII_{1,s,d} = \frac{IS_{1,s,d}}{CFS_{2,s,d}}\frac{CFS_{2,s,d}}{IS_{1,s,d}} = \frac{\delta^{P}_{0,s,1}}{\delta^{P}_{0,s,2}}$$

$$PII_{2,s,d} = \frac{IS_{2,s,d}}{CFS_{1,s,d}}\frac{CFS_{1,s,d}}{IS_{2,s,d}} = \frac{\delta^{P}_{0,s,2}}{\delta^{P}_{0,s,1}}$$

2.4 Price Discovery Efficiency ($\mathrm{PDE_{v,s,d}}$)

2.4.1 ***PDE*** is the ratio of transitory shocks in venue V_i to the competing venue(s)

2.4.2 Higher PDE in V_i reflects lower price discovery efficiency in the competing venue.

2.4.3 The formula for ***PDE*** is

$$PDE_{1,s,d} = \left|\frac{CFS_{1,s,d}}{CFS_{2,s,d}}\right| = \left|\frac{\delta^{T}_{0,s,2}}{\delta^{T}_{0,s,1}}\right|$$

$$PDE_{2,s,d} = \left|\frac{CFS_{2,s,d}}{CFS_{1,s,d}}\right| = \left|\frac{\delta^{T}_{0,s,1}}{\delta^{T}_{0,s,2}}\right|$$

	2.5 Information Leadership Share ($ILS_{v,s,d}$) 2.5.1 ***ILS*** measures the relative impact of permanent shocks on the prices series in each venue. 2.5.2 An ILS greater than 0.5 in venue ***Vi*** implies that price series in ***Vi*** leads the price discovery process after the arrival of new fundamental inforamtion . 2.5.3 The formula for ***ILS*** is $$ILS_1 = \frac{\left|\frac{IS_1}{IS_2}\ \frac{CFS_2}{CFS_1}\right|}{\left|\frac{IS_1}{IS_2}\ \frac{CFS_2}{CFS_1}\right| + \left|\frac{IS_2}{IS_1}\ \frac{CFS_1}{CFS_2}\right|}$$ $$ILS_2 = \frac{\left|\frac{IS_2}{IS_1}\ \frac{CFS_1}{CFS_2}\right|}{\left|\frac{IS_2}{IS_1}\ \frac{CFS_1}{CFS_2}\right| + \left|\frac{IS_1}{IS_2}\ \frac{CFS_2}{CFS_1}\right|}$$ **3.0 The following fields will be stored in the database at security level:** 1) Date 2) Security 3) Venue 4) PDE 5) ILS 6) PII 7) IS **8)** CFS
Benchmark	None
Definitions	1) The subscript of data arrays referred in this document are • *s* stands for security; • *v* stands for venue • *d* stands for trading day • $\delta^P_{v,s,d}$ and $\delta^T_{v,s,d}$ stands contemporaneous responses of price to the informational and frictional innovations, respectively. 2) ``All equity securities" refers all equity securities on primary venue 3) ``Cross traded securities" refers to common equity securities traded on all ATS venues

Appendix

1. Cointegration

A single asset listed in multiple venues share the same fundamental information. Hence we could assume that trade prices in each venue are cointegrated. Mechanical Forex illustrates a good example to explain Cointegration in practice.

> *"…Typical example to explain cointegration talks about a man who goes to a bar with his dog. After getting drunk and leaving the bar both the man and the dog walk the same path home, although their stochastic drift – which is the random way in which the man walks and the dog wonders along the way – are different. When this happens their paths are in fact correlated but they are not cointegrated. If the man instead decides to put a leash on the dog their paths become cointegrated because they now share a common stochastic drift that is determined by the length of the leash. The man and the dog cannot be separated further than their leash allows, which makes any random movements they make beyond a certain length common to both (as they would pull on each other). In statistics we can evaluate for cointegration using Johansen test to confirm Cointegration…"*
> (http://mechanicalforex.com/2014/11/cointegration-in-the-forex-market.html)

2. Permanent and transitory shocks

2.1 Permanent shock ($\eta_{s,t}^{P}$) is the innovation to the fundamental value reflecting new information and economic considerations.

2.2 Transitory microstructure shocks ($\eta_{s,t}^{T}$) arise from trading-related frictions, such as bid-ask bounce, rounding effects, and lagged adjustments to new information etc..

3. Structural Price Discovery Cointegration model

- This model is easy to identify permanent and transitory shocks for price series in arbitraged linked markets with minimal restrictions.
- Assume that trade price at time t (p_t) is cointegrated at order 1, and the price changes Δp_t are integrated at order zero.
- Assume Δp_t has a bivariate moving average shown in Equation (1):

$$\Delta p_t = \Psi(L)\, e_t = e_t + \Psi_1 e_{t-1} + \Psi_2 e_{t-2} + \ldots, \quad (1)$$

$$\begin{pmatrix} \Delta p_{1,s,t} \\ \Delta p_{2,s,t} \end{pmatrix} = \begin{pmatrix} \delta_{1,s,t}^{P}(L) & \delta_{1,s,t}^{T}(L) \\ \delta_{2,s,t}^{P}(L) & \delta_{2,s,t}^{T}(L) \end{pmatrix} \begin{pmatrix} \eta_{s,t}^{P} \\ \eta_{s,t}^{T} \end{pmatrix} \quad (2)$$

$$e_{1,s,t} = \delta_{1,s,t}^{P}\eta_{s,t}^{P} + \delta_{1,s,t}^{T}\eta_{s,t}^{T} \quad (3)$$

$$e_{2,s,t} = \delta_{2,s,t}^{P}\eta_{s,t}^{P} + \delta_{2,s,t}^{T}\eta_{s,t}^{T} \quad (4)$$

$$\Delta = \delta^{P}_{1,s,t}\delta^{T}_{2,s,t} - \delta^{T}_{1,s,t}\delta^{P}_{2,s,t}$$

(5)

where (v = 1, 2) is the reduced form forecasting errors.

- In both Equation (3) and (4), each of the forecasting errors can be attributed to the unobservable informational innovation (shocks) and transitory innovation (shocks).
- The intuition behind the error-correction model is that a change in price is determined by the deviation from the long-term equilibrium relationship (i.e., security price in each venue should be no different in the long-run). For the same security, cointegration analysis provides evidence of a long-run relationship between the venues.