CBOE*Direct* Configuration
Version 2.5

Feb 14th, 2002

Prepared By
Ravi Vazirani

Table of Contents

Introduction

This document describes the current configuration for the Screen-Based Trading system.

The information describes all the processes, terminology, session descriptions, cluster configuration and other such information.

Conventions Used in this Document

To determine what sections of this document have changed since its last revision, revision bars are used in the margins to show paragraphs or sections of the document that have changed.

The notation below is used to indicate important information associated with a particular function.

Note This is an example note.

Change Notices

The following change notices are provided to assist users of this document in determining the impact of current changes.

Date	Change Description
3/16/01	New document.
2/14/2002	General cleanup and additions for Stock Futures processes.
6/13/2002	Added new connections for PDS.

SBT Terminology

This section provides a list of SBT terms and descriptions that are used throughout the document.

SBT Name	Description
GlobalCluster	Comprises of 2 hosts one setup as primary and one setup as a backup. ***(Each host is also known as a 'Global Server')***. All components common to the SBT system run on these servers.
BusinessCluster	Comprises of 2 hosts one setup as primary and one setup as a backup. ***(Each host is also known as a 'Business Server').*** The business server runs the trading engine and is responsible for accepting orders, quotes and executing trades. **Note** A class (e.g. IBM) can be traded only one business cluster. But at the same time classes can be distributed across multiple clusters for load balancing/outage isolation reasons.
FrontEnd	The front end is used for routing messages (received by the CAS) to appropriate business server and to the global server.
TradingSession	The Sbt system can handle multiple sessions. Each session has a start time and end time associated with it. The Sbt system configures trading classes in sessions. Currently there are 3 sessions configured W_AM1(Also known as the ETH session) , W_MAIN(also known as the External Session, TPF Session) and ONE_MAIN (for Stock futures) Example IBM can be configured in W_AM1 and W_MAIN sessions. If a firm submits an order for W_MAIN session then the order will be routed to TPF. If the order is submitted for the W_AM1 session then the order will be routed to SBT system.
TradingClasses,Classes	Name of the trading symbol as defined by TIPS. All derivatives are a part of a trading class. E.g.. Trading class = IBM (contains derivatives IBQ,IBZ,LIB)
TradingProduct,Products	The specific option series.
RoutingGroups	In addition to TradingClasses being configured in sessions, the classes are also configured in routing groups. ***The routing group basically defines which Business cluster manages what set of products***. (e.g. RoutingGroup Name = TradeServer1 on BusinessCluster1 Assigned Classes = OEX,DJX,RUT) (e.g. RoutingGroup Name = TradeServer2 on

	BusinessCluster2 Assigned Classes = IBM)
Global Server Terminal	The login ‘`xterm`’ terminal used for manual commands.
Business Server Terminal	The login ‘`xterm`’ terminal used for entering manual commands.
Front end Terminal	The login ‘`xterm`’ terminal used for entering manual commands.
SA GUI,SysAdmin Gui	SBT’s System administration Gui used for SBT’s Maintenance functions and for Control functions on TradingSessions,User Maintenance, Performing product state changes,end of session, end of day, trade busts etc.
CAS	Client Application Services – The remote application that runs on the client side and communicates with the SBT system. Also known as the remote caching server.

Tradeengine Application Process Description

The Screen-Based Trading system is comprised of Global Clusters, Business Clusters , Front ends and CAS's. The following sections describe the processes that run on each cluster.

Tradeengine Process Layout

Global Cluster (prdgc01a & prdgc01b)

Processes	Affected Sessions
Id Server	All Sessions
Global Server	All Sessions
Membership Adapter	All Sessions
SMS Server	All Sessions
Frontend Server	All Sessions
Tips Adapter	All Sessions
Tpf Global Adapter	All Sessions (Needed at End Of Day)
Compass Adapter	ONE_MAIN
Copp Global Adapter	ONE_MAIN

Business Cluster 1 (prdbc01a & prdbc01b)

Processes	Affected Sessions
TradeServer1	W_AM1
ExternalTradeServer1	W_MAIN
Status Server	W_AM1 & W_MAIN
TpfAdapter	W_AM1 & W_MAIN
CtmAdapter	W_AM1

Business Cluster 2 (prdbc02a & prdbc02b)

Processes	Affected Sessions
TradeServer2	ONE_MAIN
Status Server	ONE_MAIN
Ctm Adapter	ONE_MAIN
Copp Adapter	ONE_MAIN

Frontend (prdfe01,prdfe02)

Processes	Affected Sessions
Frontend	All Sessions

Description of processes on Global Cluster

The Global Cluster consists of the following host's `'prdgc1a'` and `'prdgc1b'`. One is configured as 'Primary' and the other as 'Slave'. Each process on the GlobalCluster is described below

Note The explanation below will help understand what would happen if a particular process does not start or aborts intraday.

Global Server

- The Global Server process maintains the database for all products (including applying price adjustments), all users and firms from membership, all session configuration and all system wide trading properties.
- Session Management (Start of session, End of Session and End of Day) and end of day processing is performed/triggered using this process. (Global server sends events for start session,opening products,closing products,ending sessions,generating nothing dones to all the TradeServers, the TradeServer's then take appropriate action upon receiving those events).
- This process is also used for loading open interest data, updating summary information and end of day and disseminating settlement prices and summaries to SIAC.
- SAGUI requires this process for management of the system and updating global data like session information, user information, product definitions etc.
- Intraday series add also occur through the process.
- In addition, all product configurations (i.e. routing tables for clusters) are managed through this process).

Note If this process does not come up, the rest of the system will not start and all of the above will get affected.

IDServer

The id server is responsible for the following

- All records in the system have an ID that uniquely identifies the row . The id server is process is responsible for generating this unique database ids for the whole system.
- It is also responsible for generating unique ids for CTM matched sequence numbers.

Note If this process does not come up, the rest of the system will not start.

TpfGlobal

The TpfGlobal process is responsible for

- Sending Nothing Dones to TPF for orders routed by TPF to the ETH Session.
- Responsible for processing VSDL downloads from TPF.
- Processes intraday series add messages and forwards them the to the GlobalServer.
- Responsible for receiving price adjustments & order adjustment downloads messages from TPF and forwarding them to the GlobalServer for processing.

TipsAdapter

This process is responsible for

- Receiving Underlying Recap/Ticker and disseminating it to the rest of system (e.g. the CAS's/TradeServer's pick up this underlying data and process them accordingly).
- Processing Product Downloads (The Global server requests product download from TIPS to get the 'underlying' to 'alias' mapping.
- Receiving Options Best Of The Rest messages from TIPS and disseminates this information to the rest of the system for further processing.
- Receives TPF's quotes, TPF's market condition updates , TPF's Last sale/price reports and NBBO information from PDS (for open outcry products) basically the same information you see on a RCN terminal.

SMSServer

This process is also known as the Session Management Server. It is responsible for

- Maintaining user sessions (i.e. maintains relationship between all logged in users and the CAS's over which they are logged in).
- When a CAS user logs in , CAS registers the user with this process.
- When a CAS process dies the SysManAgents on the CAS box notify the SMS server that a CAS processes is stopped. Upon receiving this notification the SMS Server will notify the TradeServer to take down quotes for all those users logged in through that CAS process.

Note If this process is down, no new CAS user will be able to login to the SBT system and if a user logs out, the quotes for that user will not be taken down.

MemAdapter

The MemAdapter process is responsible for

- Receiving membership updates for firms and users (and then in turn updates the GlobalServer process).
- Receives MPP data for Market maker and DPM class assignments (and then in turn updates the Global server process accordingly).

- This program reads Membership & MPP system data every five minutes and updates the SBT system of any new changes received from the membership system.

CompassAdapter

The Compass adapter process is responsible for

- Receiving orders for FUTURES products currently and forwarding them to the appropriate TradeServer process for trading purposes.
- Receives FILLS,CANCEL REPORTS etc. from the system and forwards them to the Compass host over which the original order came on. If the Compass host is down the system will switch to the backup Compass host and route the responses to that host instead.

Note If this process is down then no orders/fills will be received from or sent back to the COMPASS firms.

CoppGlobalAdapter

Th CoppGlobalAdapter is responsible for

- Sending settlement prices for FUTURES products to COPP. (This process will query the global server to get the settlement prices and disseminate them to SIAC via COPP).
- Sending Summary for FUTURES products to COPP. (This process will query the global server to get the summary information and disseminate them to SIAC via COPP).

Note If this process is down then summaries and settlement prices cannot be disseminated to SIAC (via COPP).

Frontend Server

The Frontend server on the global cluster is used by the SA CAS / SA GUI to connect and send messages to CBOE*Direct.*

Description of process on Business Cluster

Currently in production there are 2-business clusters configured prdbc01 & prdbc02.

- Prdbc01 consists of hosts prdbc01a & prdbc01b (one host is running in master mode and the other in slave mode). ***This cluster deals with options products in W_AM1 & W_MAIN sessions.***
- Prdbc02 consists of hosts prdbc02a & prdbc02b (one host is running in master mode and the other in slave mode). ***This cluster is configured to deal with Stock Futures products in ONE_MAIN session.***

Each application process that runs on this Business cluster is explained below.

The explanation below will help understand what would happen if a particular process is not running.

TradeServer1 (Configured for session: W_AM1)

TradeServer1 is the **main trading engine** process responsible for trading options products in the ETH(W_AM1) session.

This process is responsible for accepting orders, quotes and executing trades. It is also responsible for generating all the trade data that goes along with the trading.

TradeServer2 (Configured for session: ONE_MAIN)

TradeServer1 is the main trading engine process responsible for trading FUTURES products in the ONE_MAIN session.

ExtTradeServer1 (Configured for session: W_MAIN)

The ExtTradeServer is primarily responsible for routing orders to TPF (W_MAIN session). An order that comes in for the W_MAIN session is routed to this process by the frontend. This process in turn will forward the order to TPF (via the TPF adapter running on that cluster).

When TPF fills the order , a fill report is sent by TPF to the TPF Adapter, which in turn will forward the FILL to this process.

Once this process receives the fill, it will generate a fill report that is sent to the CAS user who originally submitted the order.

Note If this process is not running, the user will not be able to route orders to TPF for trading in the W_MAIN session.

StatusServer (Configured for W_AM1,W_MAIN & ONE_MAIN sessions)

The Status server is responsible for

- Sending out Order statuses (fills, cancels, nothing dones) and Quote Statuses (fills, cancels etc) to the firms and users logged into the SBT system.
- Is also responsible for keeping track of if the user has received the status message.(This is done by the CAS acking the message stating that the status has been delivered to the user).
- Responsible for sending all unacked status messages to the user when the user logs in to the system.

TPFAdapter (Configured for W_MAIN and W_AM1 sessions)

This process is responsible for:

- Sending orders to TPF and receiving fills from TPF for the W_MAIN session. (via the ExtTradeServer1 process).
- Receiving orders from TPF and sending fills to TPF for ETH Session (via the TradeServer1 session).
- Sending Last Sales and Current Market(Best Quotes) to TPF (for OPRA reporting) for the ETH Session.
- Provides mechanism to the CAS user to service query requests for TPF MDR Service.

CtmAdapter (Configured for W_AM1 & ONE_MAIN sessions)

This process is responsible for

- Sending matched trades to CTM for the ETH (W_AM1 session).
- Sending matched trades to CTM for the FUTURES (ONE_MAIN session).

CoppAdapter (Configured for ONE_MAIN session)

This process is responsible for:

- Sending Last Sales and Current Market(Best Quotes) to COPP for SIAC reporting.

Description of processes on Frontend's

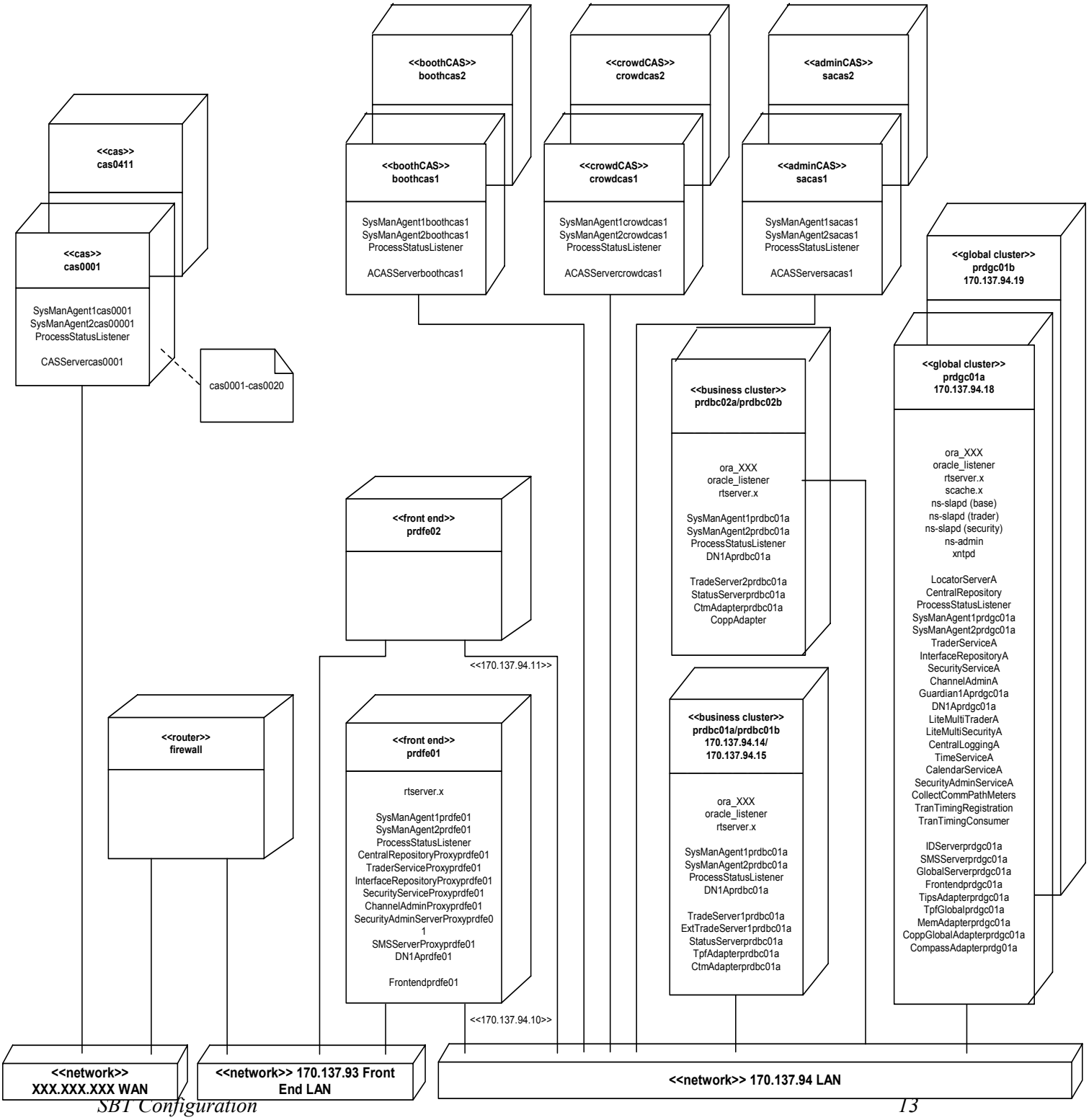
This is not a cluster configuration. The front ends consist of the hosts `'prdfe01'` and `'prdfe02'`. The processes listed below run on all front end hosts.

Frontend (Configured for ALL sessions)

This process is primary responsible for forwarding requests (e.g. orders,quotes, queries etc) entered by CAS users for a particular session to the appropriate `'TradeServer'` handling the product.

SBT Deployment Diagram

The following diagram depicts the Screen-Based Trading clusters and processes.



SBT Processes – Dependencies/Startup scripts

The table below address dependencies, startup scripts, run modes and logins for the SBT processes.

Base processes	Notes
ora_XXX	Oracle base processes (Global, Business)
oracle listener	(Global, Business)
rtserver.x	Talarian RTServer (Global, Business, and FrontEnd)
scache.x	Talarian SmartCache (Global Cluster only)
ns-slapd	netscape ldap server (3 are required - Base, Trader, Security: Global only)
ns-admin	netscape ldap server admin (Global only)
Xntpd	slave time server (Global only)

Global Cluster (Prim) java processes	Notes	Dependency	Startup Script (run dir* or VOB)	Run Mode	Login
LocatorServerA	1 Implementation Repository	Oracle	startLocator	DU	infra
CentralRepository	2	Oracle, Locator	startSysManRepository	1	infra
ProcessStatusListener	2a writes status file for Patrol		startProcessListener	IND	infra
SysManAgent1<hostname>	3a	CentralRepository, Locator	startSysManAgent	IND	infra
SysManAgent2<hostname>	3b	Locator	startSysManAgent -B	IND	infra
TraderServiceA	4	ns-slapd (TraderService), Locator	startTrader	DU	infra
InterfaceRepositoryA	5 Interface Repository	Oracle, Locator	startIFR	DU	infra
SecurityServiceA	6	ns-slapd (SecurityService), Trader, Locator	startSecurity	DU	infra
ChannelAdminA	7	TraderService, Locator	startChannelAdmin	DU	infra
Guardian1A<hostname>	8 DN Guardian		startGuardian	DU	infra
DN1A<hostname>	9 Distribution Node	Guardian	startDN	MS	infra
LiteMultiTraderA	multi-master repl (b side)	ns-slapd (TraderService)	startLiteMulti	DU	infra
LiteMultiSecurityA	multi-master repl (b side)	ns-slapd (SecurityService)	startLiteMulti -SSecurity	DU	infra
CentralLoggingA		Oracle, Trader	startLoggingServer	DU	infra
TimeServiceA		Trader, Locator	startTImeService	DU	infra
CalendarServiceA		TimeService, Trader, Locator	startCalendarService	DU	infra
SecurityAdminServiceA		SecurityService, Trader	startSAS	DU	infra
CollectCommPathMeters		Oracle	startCCM	1	infra
TranTimingRegistration			startTimingRegistration	1	infra

TranTimingConsumer			startTimingConsumer	1	infra

IDServer<hostname>	application ID Service	Oracle	startServer id	MS	tradeeng
SMSServer<hostname>	Session Manager	SysManAgent<hostname>	startServer sms	MS	tradeeng
GlobalServer<hostname>	application global server	Oracle	startServer global	MS	tradeeng
Frontend<hostname>	application FE for Adapters		startServer frontend	DU	tradeeng
TipsAdapter<hostname>	application external adapter		startServer tips	MS	tradeeng
TpfGlobal<hostname>	application external adapter		startServer tpfGlobal	MS	tradeeng
MemAdapter<hostname>	application external adapter		startServer membership	MS	tradeeng
CompassAdapter<hostname>	application external adapter		startServer compass	MS	tradeeng
CoppGlobal<hostname>	application external adapter		startServer coppGlobal	MS	tradeeng

Global Cluster (Sec) java processes	**Notes**	**Dependency**	**Startup Script (run dir* or VOB)**	**Run Mode**	**Login**
LocatorServerB	Implementation Repository	Oracle	startLocator –B	DU	infra
CentralRepository		Oracle	StartSysManRepository	1	infra
ProcessStatusListener	writes status file for Patrol		StartProcessListener	IND	infra
SysManAgent1<hostname>		CentralRepository, Locator	StartSysManAgent	IND	infra
SysManAgent2<hostname>		Locator	startSysManAgent –B	IND	infra
TraderServiceB		ns-slapd (TraderService)	startTrader –B	DU	infra
InterfaceRepositoryB	Interface Repository	Oracle	startIFR –B	DU	infra
SecurityServiceB		ns-slapd (SecurityService)	startSecurity –B	DU	infra
ChannelAdminB		TraderService	startChannelAdmin -B	DU	infra
Guardian1B<hostname>	DN Guardian		startGuardian –B	DU	infra
DN1B<hostname>	Distribution Node		startDN –B	MS	infra
LiteMultiTraderB	multi-master replication (a side)	ns-slapd (TraderService)	startLiteMulti –B	DU	infra
LiteMultiSecurityB	multi-master replication (a side)	ns-slapd (SecurityService)	startLiteMulti -B –Ssecurity	DU	infra
CentralLoggingB		Oracle	startLoggingServer -B	DU	infra
TimeServiceB		Trader, Locator	startTImeService -B	DU	infra
CalendarServiceB		TimeService, Trader, Locator	startCalendarService -B	DU	infra
SecurityAdminServiceB		SecurityService, Trader	startSAS -B	DU	infra
CollectCommPathMeters		Oracle	startCCM	1	infra
TranTimingRegistration			StartTimingRegistration	1	infra
TranTimingConsumer			StartTimingConsumer	1	infra

IDServer<hostname>	application ID Service	Oracle	startServer id	MS	tradeeng
SMSServer<hostname>	Session Manager	SysManAgent<hostname>	startServer sms	MS	tradeeng
GlobalServer<hostname>	application global server	Oracle	startServer global	MS	tradeeng
Frontend<hostname>	application FE for Adapters		startServer Frontend	DU	tradeeng

TipsAdapter<hostname>	application external adapter		startServer tips	MS	tradeeng
TpfGlobal<hostname>	application external adapter		startServer tpfglobal	MS	tradeeng
MemAdapter<hostname>	application external adapter		startServer Membership	MS	tradeeng
CoppGlobal<hostname>	application external adapter		startServer coppGlobal	MS	tradeeng
CompassAdapter<hostname>	application external adapter		startServer compass	MS	tradeeng

Business Cluster (Prim) java processes	**Notes**	**Dependency**	**Startup Script (run dir* or VOB)**	**Run Mode**	**Login**
SysManAgent1<hostname>		CentralRepository, Locator	StartSysManAgent	IND	infra
SysManAgent2<hostname>		Locator	startSysManAgent –B	IND	infra
ProcessStatusListener	writes status file for Patrol		StartProcessListener	IND	infra
DN1A<hostname>	Distribution Node		StartDN	MS	infra
TradeServer1<hostname>	application business server	Oracle	startServer trade –n1	MS	tradeeng
ExtTradeServer1<hostname>	application business server	Oracle	startServer external -n1	MS	tradeeng
StatusServer<hostname>	application business server	Oracle	startServer status	MS	tradeeng
TpfAdapter<hostname>	application external adapter		startServer tpf	MS	tradeeng
CtmAdapter<hostname>	application external adapter		startServer ctm	MS	tradeeng
CoppAdapter<hostname>	application external adapter		startServer copp	MS	tradeeng
TradeServer2<hostname>	application external adapter		startServer trade –n2	MS	tradeeng

Business Cluster (Sec) java processes	**Notes**	**Dependency**	**Startup Script (run dir* or VOB)**	**Run Mode**	**Login**
SysManAgent1<hostname>		CentralRepository, Locator	StartSysManAgent	IND	infra
SysManAgent2<hostname>		Locator	startSysManAgent –B	IND	infra
ProcessStatusListener	writes status file for Patrol		StartProcessListener	IND	infra
DN1B<hostname>	Distribution Node		startDN –B	MS	infra
TradeServer1<hostname>	application business server	Oracle	startServer trade –n1	MS	tradeeng
ExtTradeServer1<hostname>	application business server	Oracle	startServer external	MS	tradeeng
StatusServer<hostname>	application business server	Oracle	startServer status	MS	tradeeng
TpfAdapter<hostname>	application external adapter		startServer tpf	MS	tradeeng
CtmAdapter<hostname>	application external adapter		startServer ctm	MS	tradeeng
CoppAdapter<hostname>	application external adapter		startServer copp	MS	tradeeng
TradeServer2<hostname>	application external adapter		startServer trade –n2	MS	tradeeng

Front End java processes	**Notes**	**Dependency**	**Startup Script (run dir* or VOB)**	**Run Mode**	**Login**
SysManAgent1<hostname>		CentralRepository, Locator	StartSysManAgent	N/A	infra
SysManAgent2<hostname>		Locator	startSysManAgent -B	N/A	infra

ProcessStatusListener	writes status file for Patrol		startProcessListener	N/A	infra
CentralRepositoryProxy<hostname>	Proxy for Central Repository	CentralRepository	startSysManRepositoryProxy	N/A	infra
TraderServiceProxy<hostname>	Proxy for TraderService	TraderService	startTraderProxy	N/A	infra
InterfaceRepositoryProxy<hostname>	Proxy for InterfaceRepository	InterfaceRepository	startIFRProxy	N/A	infra
SecurityServiceProxy<hostname>	Proxy for SecurityService	SecurityService	startSecurityProxy	N/A	infra
ChannelAdminProxy<hostname>	Proxy for ChannelAdmin	ChannelAdmin	startChannelAdminProxy	N/A	infra
SMSServerProxy<hostname>	Proxy for Session Mgmt	SMSServer	startSMSProxy	N/A	infra
DN1A<hostname>	Distribution Node		startDN	MS	infra
Frontend<hostname>	application front end server		startServer Frontend	N/A	tradeeng

CAS java processes	**Notes**	**Dependency**	**Startup Script (run dir* or VOB)**	**Run Mode**	**Login**
SysManAgent1<hostname>		CentralRepository, Locator	startSysManAgent	N/A	infra
SysManAgent2<hostname>		Locator	startSysManAgent -B	N/A	infra
ProcessStatusListener	writes status file for Patrol		startProcessListener	N/A	infra
CASServer<hostname>	Client Application Server		startCas	N/A	tradeeng

Admin CAS java processes	**Notes**	**Dependency**	**Startup Script (run dir* or VOB)**	**Run Mode**	**Login**
SysManAgent1<hostname>		CentralRepository, Locator	startSysManAgent	N/A	infra
SysManAgent2<hostname>		Locator	startSysManAgent -B	N/A	infra
ProcessStatusListener	writes status file for Patrol		startProcessListener	N/A	infra
ACASServer<hostname>	Admin Client Application Server		startACas	N/A	tradeeng

*** run dir == $RUN_DIR environment variable**

Connections to External Systems

This section describes the connections to the external systems.

TipsAdapter (on Global Cluster)

On the primary `Global Server` there will be three connections established to TIPS.

Connection Name	Description
`TipsUnderlyingUdpAConnection`	Connection to the 'A' global backbone for receiving Tips broadcast data.
`TipsUnderlyingUdpAConnection`	Connection to the 'B' global backbone for receiving Tips broadcast data.
`TipsUnderlyingConnection`	For receiving Underlying Ticker/Recap.
`TipsProductDownloadConnection`	For receiving product downloads from TIPS.
`PdsMarketDataConnection`	For receiving market data information from PDS for open outcry products. The information received is TPF's quotes ,last sale, market condition updates and NBBO information.

TpfGlobalAdapter(on Global Cluster)

On the primary `Global Server` there will be two host gateway connections established to TPF.

Connection Name	Description

`TpfControlServiceConnection`
`Sbt Name = SBT01`
`Tpf Name = CONTROL`
	Used for receiving intraday series add, price adjustments and sending Nothing Dones to TPF.

`TpfVsdlServiceConnection`
`Sbt Name = SBT01`
`Tpf Name = VSDL`
	Used for VSDL class downloads from TPF.

CoppGlobalAdapter(on Global Cluster)

The following connections will be established to COPP.

Connection Name	Description

`CoppGlobalServiceConnection`
`Sbt Name = CBOEDIRC`
`CoppName = COPPFUT`
	Used for disseminating settlement prices, End of day summaries

CompassAdapter (on Global Cluster)

The following connection will be established on the primary business server.

Connection Name	Description
M8CompassOrderServiceConnection Sbt Name = SBTORD8 Compass Name = CMPORD8	Used for receiving orders from COMPASS.
M9CompassOrderServiceConnection Sbt Name = SBTORD9 Compass Name = CMPORD9	Used for receiving orders from COMPASS.
M8CompassReportsServiceConnection Sbt Name = SBTREP8 Compass Name = CMPREP8	Used for sending fills, cancel reports, nothing dones back to COMPASS.
M9CompassReportsServiceConnection Sbt Name = SBTREP9 Compass Name = CMPREP9	Used for sending fills, cancel reports, nothing dones back to COMPASS.

TpfAdapter (on Business Cluster 1)

The following connection will be established on the primary business server.

Connection Name	Description
TpfMdrServiceConnection Sbt Name = SBT01 Tpf Name = MDR	Used for receiving intraday series add, price adjustments and sending Nothing Dones to TPF.
TpfBestQuoteServiceConnection Sbt Name = SBT01 Tpf Name = SBTQTE	Used for sending SBT's Current Market(BestQuotes) to OPRA and for TPF MDR Requests.
TpfOrderServiceInboundConnection Sbt Name = SBT01 Tpf Name = ORDERS	Used for receiving ETH(W_AM1 session) orders from TPF.
TpfOrderServiceOutboundConnection Sbt Name = SBT01OUT Tpf Name = ORDERS	Used for sending orders to TPF(W_MAIN session). These are orders send by firms/users to be traded in the open out cry world.
TpfReportsServiceInboundConnection Sbt Name = SBT01 Tpf Name = REPORTS	Used for receiving reports (fills, cancels, TLTC's, Nothing Dones etc) from TPF for orders submitted by SBT to TPF.
TpfReportsServiceOutboundConnection Sbt Name = SBT01OUT Tpf Name = REPORTS	Used for sending reports(fills,cancels,TLTS'c) to TPF for orders submitted by TPF to the SBT system.
TpfTagServiceConnection Sbt Name = SBT01 Tpf Name = TAGDATA	Used to tag TPF for ORDERS and REPORTS services on TPF. Since the trading classes are distributed by cluster , TPF needs to be tagged so they know which cluster to route the orders and fills to.
TpfLastSaleServiceConnection Sbt Name = SBT01 Tpf Name = LASTSALE	Used for sending SBT's Current Market(BestQuotes) to OPRA and for TPF MDR Requests.

CoppAdapter(on Business Cluster 2)

The following connections will be established to COPP.

Connection Name	Description
CoppLastSaleServiceConnection Sbt Name = CBOEDIRL CoppName = COPPFUT	Used for disseminating last sale information for trades on Stock Futures products to SIAC.
CoppCurrentMarketServiceConnection Sbt Name = CBOEDIRQ CoppName = COPPFUT	Used for disseminating Best Quote information,Current market conditions for Stock Futures products to SIAC.

CtmAdapter(on Business Cluster 1)

The following connections will be established to CTM.

Connection Name	Description
CtmOptionsReportsServiceConnection Sbt Name = SB01SB01 CTM Name = CTMINPUT	Used to sending **OPTIONS** matched trade reports generated on CBOE*Direct* to CTM for clearing purposes.
CtmOptionsAcknowledgementsServiceConnection Sbt Name = SB01SB01 CTM Name = CTMOUTPT	Used for receiving acknowledements for **OPTIONS** matched trades that are received by CTM.

CtmAdapter(on Business Cluster 2)

The following connections will be established to CTM.

Connection Name	Description
CtmFuturesReportsServiceConnection Sbt Name = FU01FU01 CTM Name = CTMINPUT	Used to sending **STOCK FUTURES** matched trade reports generated on CBOE*Direct* to CTM for clearing purposes.
CtmFuturesAcknowledgementsServiceConnection Sbt Name = FU01FU01 CTM Name = CTMOUTPT	Used for receiving acknowledements for **STOCK FUTURES** matched trades that are received by CTM.

Trading Session Configuration

Currently there are four sessions defined in the CBOE*Direct* system.

- W_AM1 (ETH) session - Used for options products traded on CBOE*Direct*.
- ONE_MAIN session - Used for stock futures products traded on CBOE*Direct*.
- W_MAIN (OpenOutCry) session - Used for products traded on TPF.
- Underlying session - This session is needed for SBT users to get the underlying ticker and recap received from TIPS.

The configuration below lists the times and other information associated with the sessions. All session times are represented in 24 hour clocks.

W_AM1 (ETH) – Session Configuration

Configuration Parameter	Value	Description
Start Session Time	04:30	Session will automatically start at this time.
Product Pre-open time	06:00	All Products in the session automatically pre-open at this time. Orders can now be accepted in the system
Product Open time	07:00	All products will automatically enter Opening rotation and then to open after the opening rotation time period has elapsed.
Product Close time	08:15	All products will automatically be closed for trading at this time. **Note** After the close and before the session is stopped the users can still submit cancel orders.
End Session time	08:30	At this time Nothing dones will be sent out and the trading session will be stopped. After this point in time no new orders/quotes can be submitted for this session.

ONE_MAIN(Stock Futures) – Session Configuration

Configuration Parameter	Value	Description
Start Session Time	04:30	Session will automatically start at this time.
Product Pre-open time	06:00	All Products in the session automatically pre-open at this time. Orders can now be accepted in the system
Product Open time	08:30	All products will automatically enter Opening rotation and then to open after the opening rotation time period has elapsed.
Product Close time	15:15	All products will automatically be closed for trading at this time. **Note** After the close and before the session is stopped the users can still submit cancel orders.
End Session time	16:00	At this time Nothing dones will be sent out and the trading session will be stopped. After this point in time no new orders/quotes can be submitted for this session.

Underlying – Session Configuration

Configuration Parameter	Value	Description
Start Session Time	04:00	Session will automatically start at this time.
Product Pre-open time	Not applicable	Product states are not changed automatically.
Product Open time	Not applicable	Product states are not changed automatically.
Product Close time	Not applicable	Product states are not changed automatically.
End Session time	16:00	Session will automatically end at this time.

W_MAIN (TPF/Open Outcry) – Session Configuration

Configuration Parameter	Value	Description
Start Session Time	04:30	Session will automatically start at this time.
Product Pre-open time	Not applicable	Product states are changed automatically when data is received from PDS.
Product Open time	Not applicable	Product states are changed automatically when data is received from PDS.
Product Close time	Not applicable	Product states are changed automatically when data is received from PDS..
End Session time	17:00	Session will automatically end at this time. No new messages for this session will be accepted after this point in time.

Note The W_MAIN is also configured as the last session for the business day. When this session closes the End Of Day processing beings (generally) automatically. (See EOD Procedures for more detailed information).

Logging in as a SBT Operator

Tradeeng Assurance

Login: tradenga

Tradeeng Production

Login tradengp

Infrastructure Assurance

Login : infraa

Infrastructure Production

Login : infrap

Symbol Distribution

Business Cluster 1 (prdbc01a & prdbc01b)

Processes	Affected Sessions	Class Assignment (based on 1st character of the symbol)
TradeServer1	W_AM1	A-Z
ExternalTradeServer1	W_MAIN	A-Z
Status Server	W_AM1 & W_MAIN	A-Z
TpfAdapter	W_AM1 & W_MAIN	A-Z
CtmAdapter	W_AM1	A-Z

Business Cluster 2 (prdbc02a & prdbc02b)

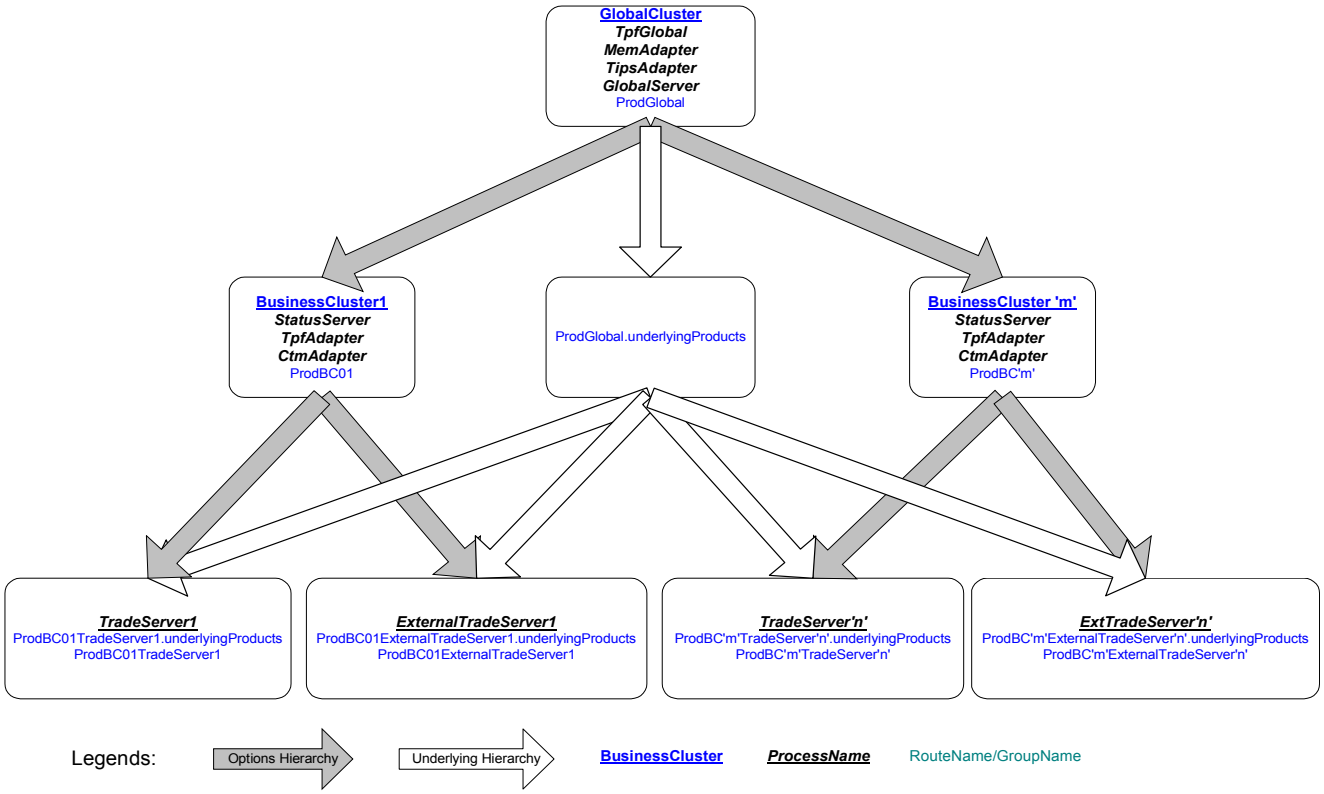
Processes	Affected Sessions	Class Assignment (based on 1st character of the symbol)
TradeServer2	ONE_MAIN	A-Z
Status Server	ONE_MAIN	A-Z
Ctm Adapter	ONE_MAIN	A-Z
Copp Adapter	ONE_MAIN	A-Z

CAS to Front End Mapping

1) All SA CAS's go through the frontend process on the GlobalServer.
2) All Trader CAS's come in through the Frontend processes running on prdfe01 & prdfe02.

Appendix A – Routing group assignment

Routing Group Hierarchy



Ravi Vazirani (03/16/2001)