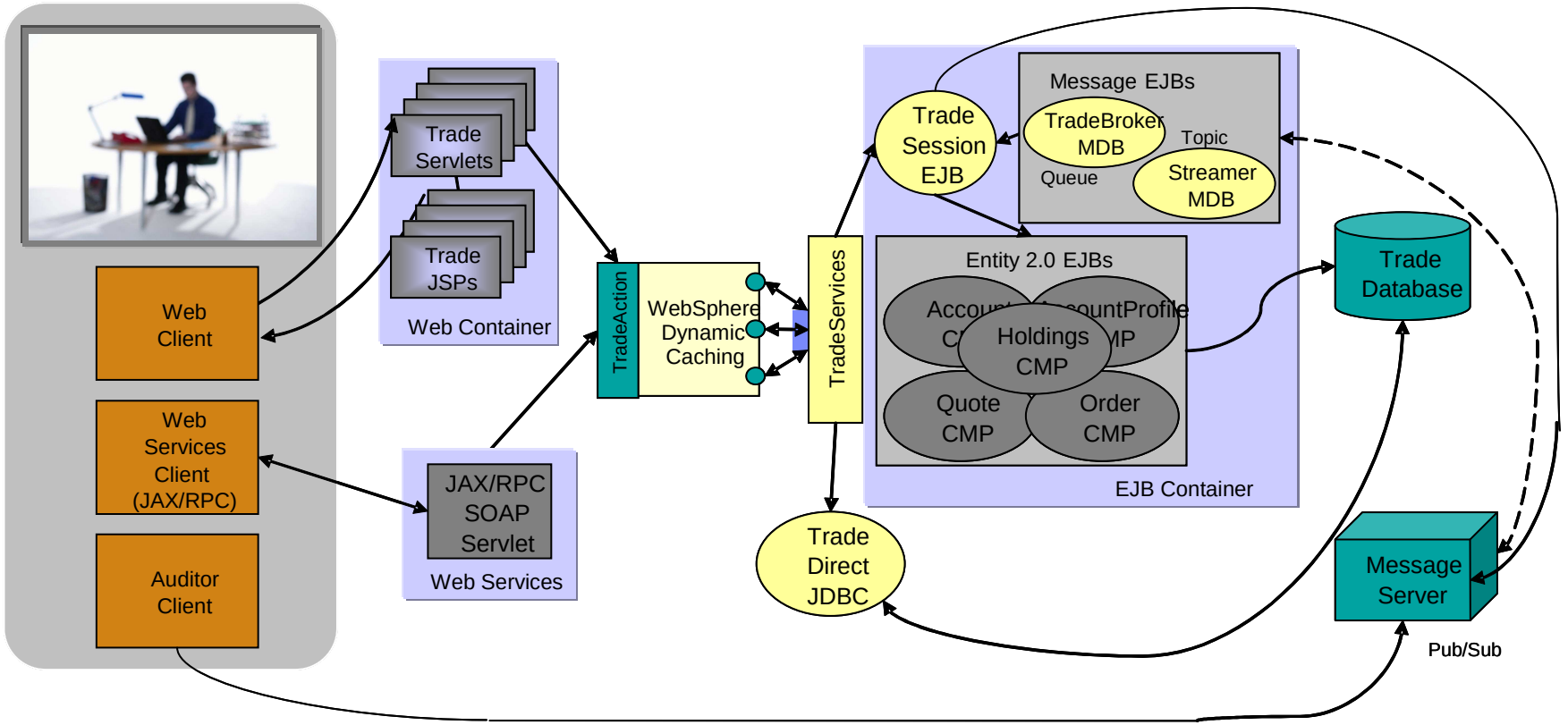
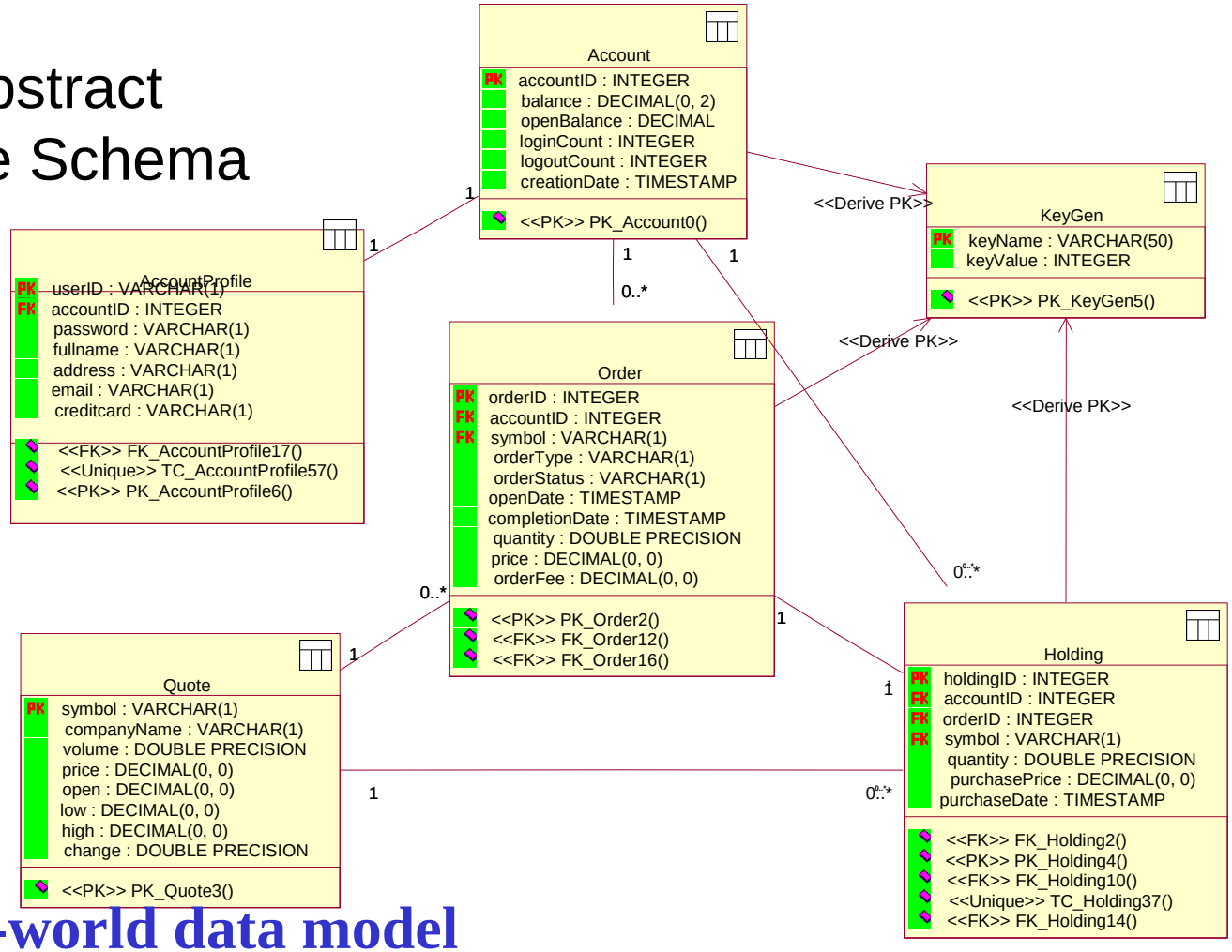
e-business software
WebSphere. software
The difference is WebSphere.
IBM

Trade6

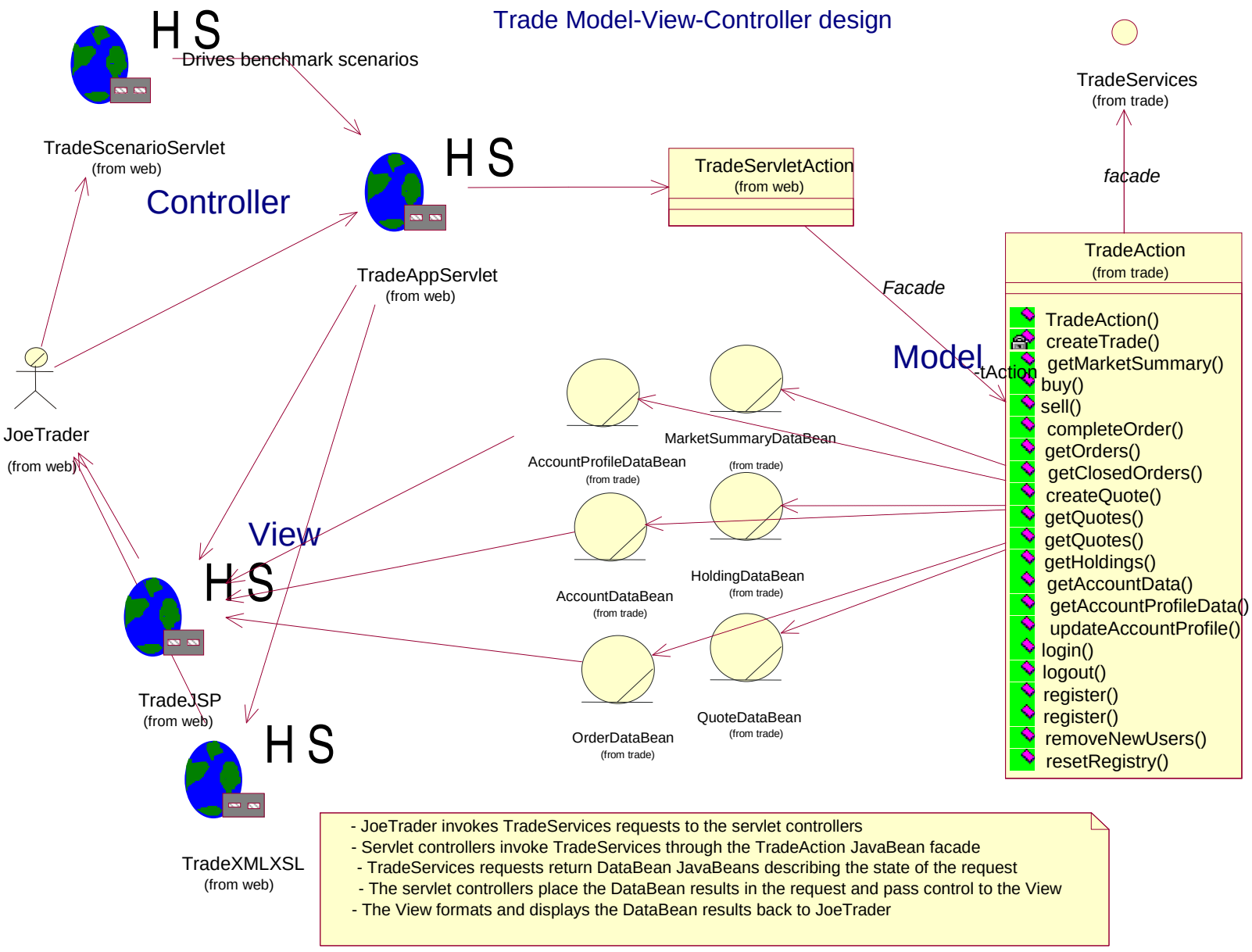


Trade UML Architecture

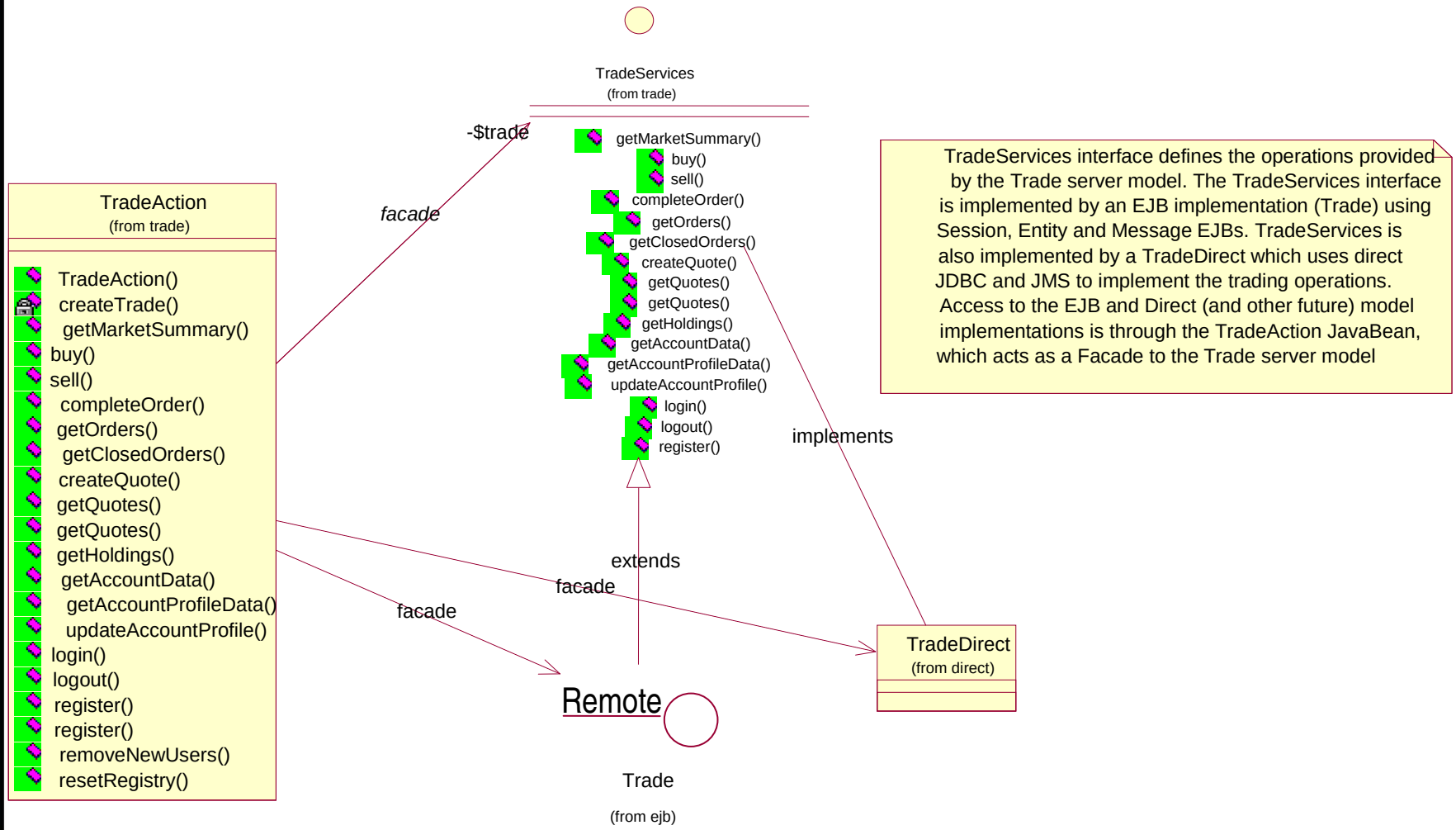
Trade Abstract Persistence Schema



- **Real-world data model**
  - Container managed rel. (CMR), RDB foreign key relationships
    - Cascading deletes, Referential integrity not enforced
  - Extended data types (BigDecimal, Timestamp, etc.)


Trade Model-View-Controller design
HS
Drives benchmark scenarios
TradeScenarioServlet
(from web)
Controller
HS
TradeAppServlet
(from web)
TradeServletAction
(from web)
TradeServices
(from trade)
facade
TradeAction
(from trade)
Facade
Model
TradeAction()
createTrade()
getMarketSummary()
buy()
sell()
completeOrder()
getOrders()
getClosedOrders()
createQuote()
getQuotes()
getQuotes()
getHoldings()
getAccountData()
getAccountProfileData()
updateAccountProfile()
login()
logout()
register()
register()
removeNewUsers()
resetRegistry()
JoeTrader
(from web)
View
HS
TradeJSP
(from web)
HS
TradeXMLXSL
(from web)
MarketSummaryDataBean
(from trade)
AccountProfileDataBean
(from trade)
HoldingDataBean
(from trade)
AccountDataBean
(from trade)
QuoteDataBean
(from trade)
OrderDataBean
(from trade)
- JoeTrader invokes TradeServices requests to the servlet controllers
- Servlet controllers invoke TradeServices through the TradeAction JavaBean facade
- TradeServices requests return DataBean JavaBeans describing the state of the request
- The servlet controllers place the DataBean results in the request and pass control to the View
- The View formats and displays the DataBean results back to JoeTrader

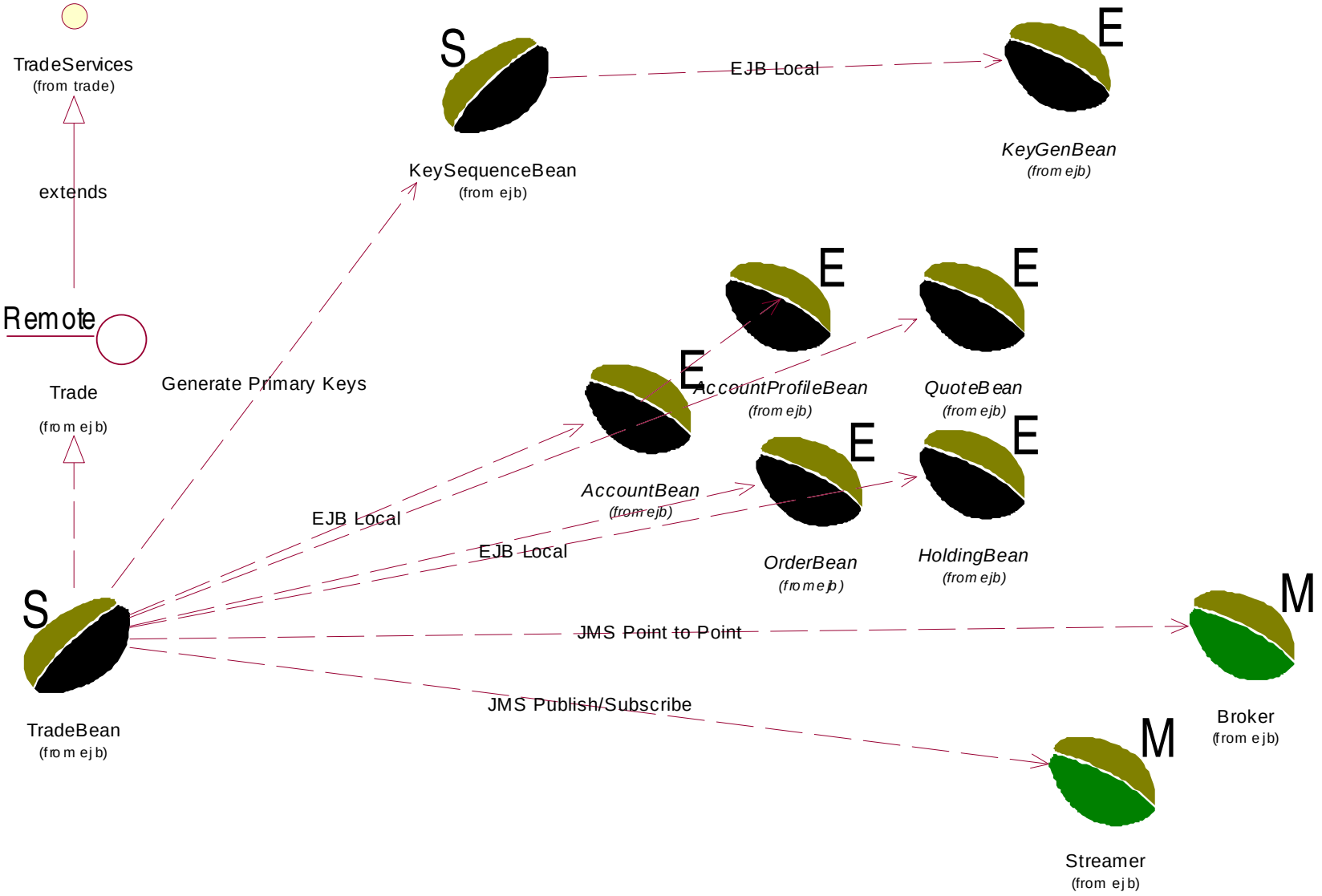
Trade Services Model Class Diagram

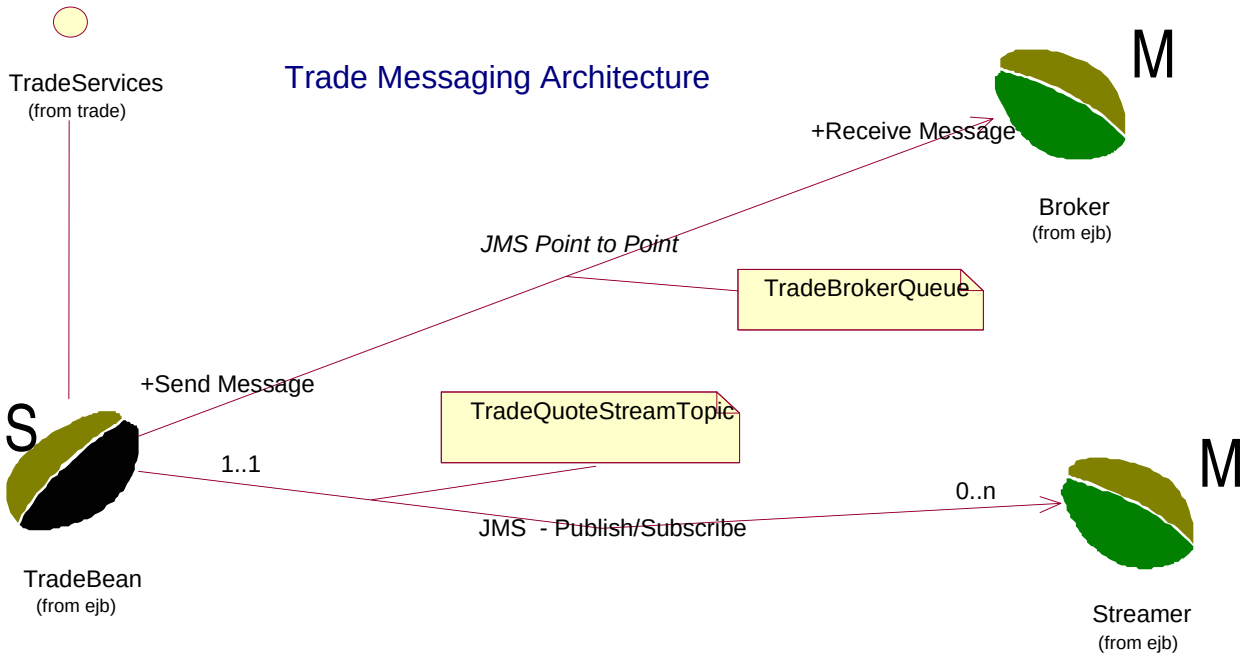


TradeServices interface defines the operations provided by the Trade server model. The TradeServices interface is implemented by an EJB implementation (Trade) using Session, Entity and Message EJBs. TradeServices is also implemented by a TradeDirect which uses direct JDBC and JMS to implement the trading operations. Access to the EJB and Direct (and other future) model implementations is through the TradeAction JavaBean, which acts as a Facade to the Trade server model

e-business software
The difference is WebSphere.
IBM

Trade EJB Model Diagram



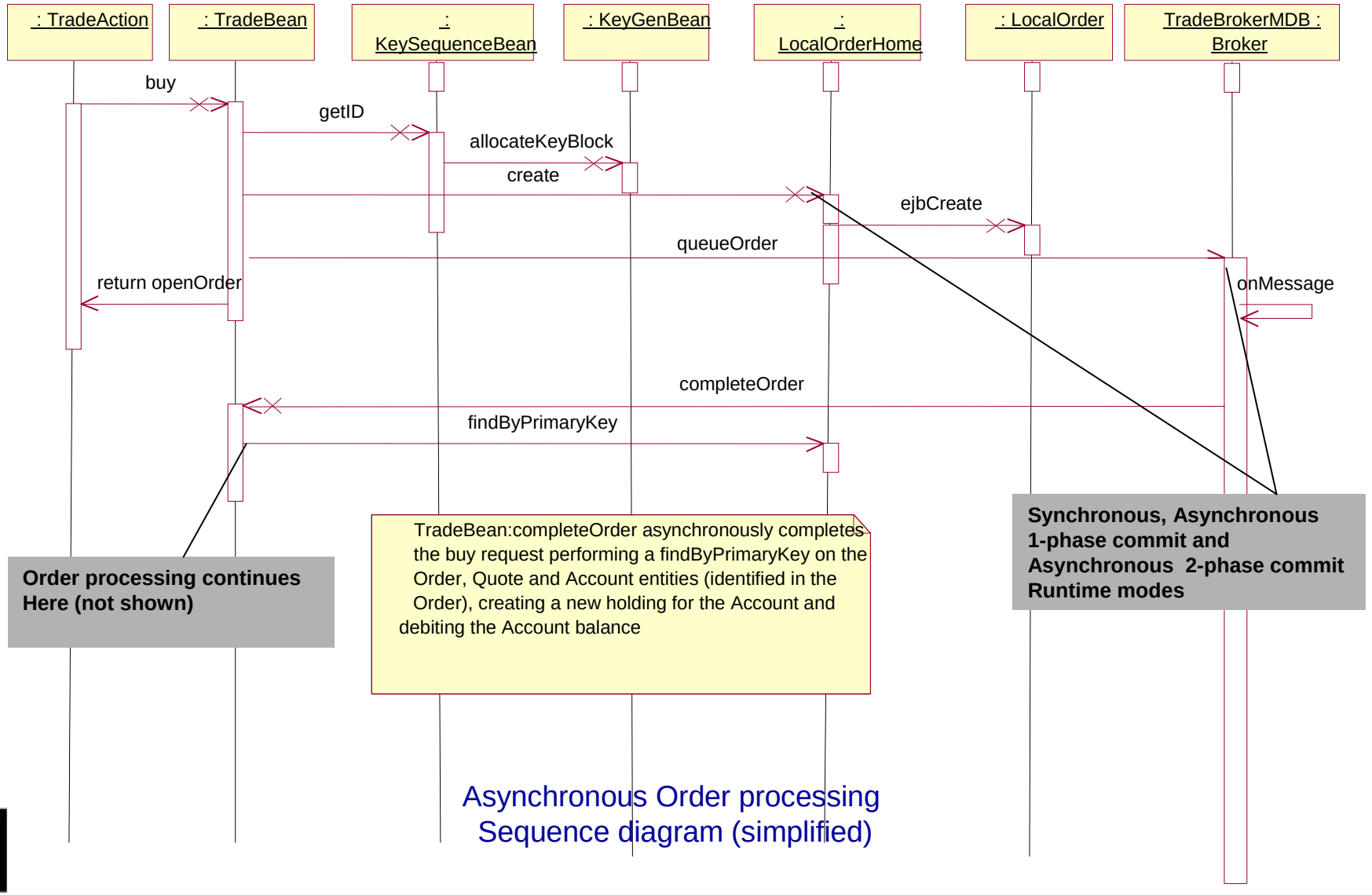


Trade leverages JMS Point to Point and Publish/Subscribe messaging

Point to Point
- Trade Order processing is asynchronous
- When a buy or sell is requested, the TradeBean Session EJB creates a new Order Entity
- The open Order information is queued to the Broker MDB for processing
- Control immediately returns to the caller with details of the newly opened (but uncompleted order)
- The Broker MDB processes the order to completion by invoking TradeServices to purchase, sell, etc. asynchronously
- Upon completion, the Order status is set to completed and the user is notified

Publish/Subscribe
- Trade incorporates a Streamer MDB which allows subscibes to be notified of Stock price changes
- When a Quote price is modified by the Trade Session EJB, a JMS message is published queued to the TradeQuoteStreamTopic
- The Streamer MDB may provide 0 or more subscribers to the Stock Quote change Topic
- All subscribers to this Topic are notified of the Stock Quote price change

e-business software
The difference is WebSphere.
IBM



Asynchronous Order processing Sequence diagram (simplified)

The difference is WebSphere.