Εννοιολογική Μοντελοποίηση Συστημάτων: Εργασία 5

Λυδία Αθανασίου 3170003

# Μεταμοντέλο

Μεταμοντέλο εφαρμογής Ιστού για τη διαβίβαση εντολών αγοράς και πώλησης μετοχών για λογαριασμό μιας χρηματιστηριακής εταιρίας, προς χρήση των πελατών που επιθυμούν να πραγματοποιούν συναλλαγές μέσω Διαδικτύου.

Στο συγκεκριμένο Μεταμοντέλο υπάρχουν δύο βασικές οντότητες, οι Entity Types και οι Process Type. Οι πρώτες αφορούν τις οντότητες του συστήματος, δηλαδή τον πελάτη, τον μετοχικό τίτλο, την εντολή που δίνει ο πελάτης στην διεπαφή και γενικότερα οποιαδήποτε οντότητα μπορεί να οριστεί σε οποιοδήποτε μοντέλο που έχει attributes και συσχετίσεις με άλλες οντότητες. Η δεύτερη αφορά όλες τις υπάρχουσες λειτουργίες του συστήματος και την αναπαράστασή τους. Το ανώτερο στρώμα του μοντέλου είναι ένα μετα-μοντέλο για τις δύο αυτές βασικές κατηγορίες στοιχείων του πεδίου (με κεντρικά στοιχεία τις μετατάξεις EntityType και ProcessType), το οποίο συνιστά το πλαίσιο για τον ορισμό των διαφόρων οντοτήτων και διεργασιών του πεδίου. Ως προς τις διεργασίες ειδικότερα, το μετα-μοντέλο είναι σύμφωνο με το γνωστό μοντέλο IDEF0 και προβλέπει την μερωνυμική ανάλυση μιας διεργασίας σε τέσσερεις διαδοχικές στάθμες. Στο μάθημα Ανάλυση και Σχεδίαση Πληροφοριακών Συστημάτων είχαμε αποδώσει συγκεκριμένο χαρακτήρα στις στάθμες αυτές και τις είχαμε ονομάσει Εφαρμογή, Σύνθετη Λειτουργία, Λειτουργία και Πράξη. Από το επίπεδο IDEF0 πλαίσιο μπορούν να προκύψουν σαν πραγματώσεις αυτού οι υπόλοιπες στάθμες πραγμάτωσης των Σύνθετων λειτουργιών, Λειτουργιών και Πράξεων. Συνεπώς το μετα-μοντέλο υποστηρίζει την περαιτέρω ανάλυση σε λειτουργίες και πράξεις όπως φαίνεται και στην παρακάτω ανάλυση. Οι Σύνθετες λειτουργίες αναλύονται σε λειτουργίες και οι λειτουργίες αναλύονται σε πράξεις . Η Α1 Σύνθετη Λειτουργία αναλύεται στις λειτουργίες Α11 και Α12( Ομοίως και η Α2 και οι Α3) και η λειτουργία Α11 αναλύεται στις πράξεις Α111, Α112 και Α113( ομοίως αναλύονται και οι υπόλοιπες λειτουργίες).

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο, χάρτης

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει πίνακας

Περιγραφή που δημιουργήθηκε αυτόματα

Σύνταξη και δημιουργία των απαραίτητων κλάσεων για το εργαλείο μοντελοποίησης ADOXX.

Εικόνα που περιέχει στιγμιότυπο οθόνης

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει στιγμιότυπο οθόνης

Περιγραφή που δημιουργήθηκε αυτόματαΕικόνα που περιέχει στιγμιότυπο οθόνης

Περιγραφή που δημιουργήθηκε αυτόματα

**Δήλωση των κλάσεων Process Type:**

GRAPHREP PEN w:0.05cm RECTANGLE x:-2cm y:-0.5cm w:4cm h:4cm

**Δήλωση των κλάσεων Entity Type:**

GRAPHREP PEN w:0.05cm ELLIPSE x:0.00cm y:0.00cm rx:2.00cm ry:0.80cm  ATTR "Name" y:-0.2cm w:c:2.8cm h:t  AVAL k:"key"  IF (k=”1”)  {   LINE x1:-1cm y1:0.2cm x2: 1cm y2:0.2cm  }

**Δήλωση των Relation Classes :**

GRAPHREP SHADOW off PEN style:solid w:0.05cm EDGE MIDDLE ATTR "control" x:0cm y:-0.6cm w:c END LINE x1:0.1cm y1:0.0cm x2:-0.2cm y2:0.1cm LINE x1:0.1cm y1:0.0cm x2:-0.2cm y2:-0.1cm

**Κώδικας στην Τελος του Μεταμοντέλου:**

BEGINTRANSACTION **TELL** Individual EntityType in M1\_Class end EntityType

**TELL** Individual Constrained in M1\_Class end Constrained

**TELL** Attribute has Attribute from: EntityType to: EntityType in M1\_Class with attribute constrained: Constrained end hasAttribute

**TELL** Individual ProcessType in M1\_Class end ProcessType

**TELL** Attribute input from: EntityType to: ProcessType in M1\_Class end input

**TELL** Attribute control from: EntityType to: ProcessType in M1\_Class end output

**TELL** Attribute mechanism from: EntityType to: ProcessType in M1\_Class end mechanism

**TELL** Attribute output from: ProcessType

to: EntityType in M1\_Class end output

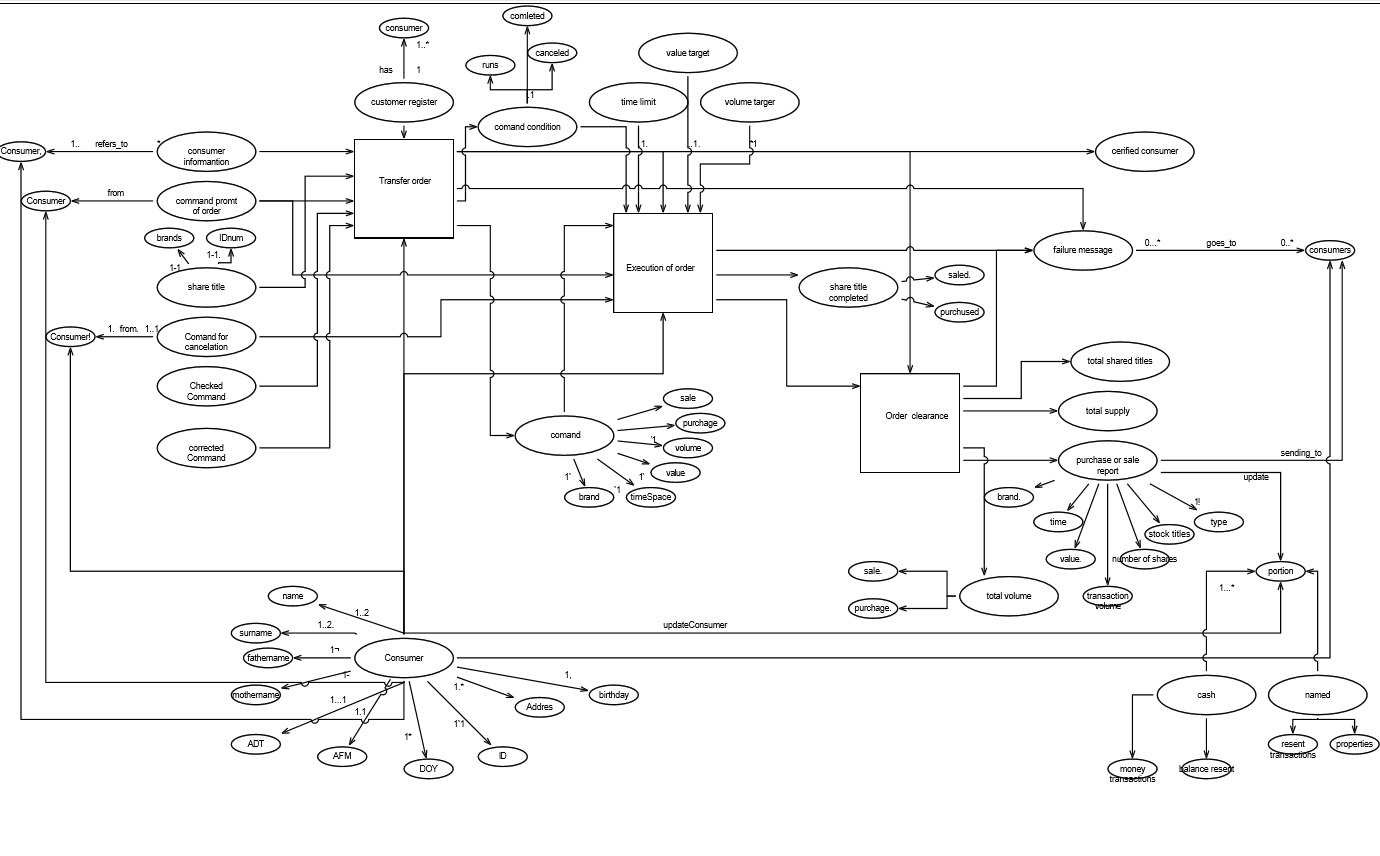
ENDTRANSACTION

\*\*Για να ορίσω τους περιορισμούς πλήθος του γραφήματος IDEF στην γλώσσα τέλος δήλωσα στο μεταμοντέλο μου τις κατάλληλες μετατάξεις περιορισμών πλήθους. Στη συνέχεια σε επίπεδο s\_class δημιούργησα τις απαραίτητες πραγματώσεις αυτών των μετατάξεων ώστε να μπορούν να ορισθούν όλοι οι απαραίτητοι περιορισμού πλήθους.

Διάγραμμα του εννοιολογικό μοντέλου της εφαρμογής, με χρήση της πλατφόρμας ADOxx

(Σύνθετες Λειτουργίες).

Το μοντέλο θα περιγράφει τόσο τις οντότητες που εμφανίζονται στην εφαρμογή, όσο και τις διεργασίες που λαμβάνουν χώρα στις Σύνθετες λειτουργίες της εφαρμογής. Παρακάτω στην γλώσσα τελος εμφανίζονται και οι κληρονομικότητες μεταξύ των οντοτήτων. Για παράδειγμα η εντολή ακύρωση και η εντολή κατάστασης είναι isA της εντολής καθώς κληρονομούν τα γνωρίσματά της και τις ιδιότητές της και έχουν η κάθε μία κάποια επιπλέον γνωρίσματα και ιδιότητες.



\*\* η εικόνα του διαγράμματος καθώς και ο κώδικας της τέλος (αρχέιο .tls) βρίσκονται στον φάκελο της εργασίας για μεγαλύτερη ευκρίνεια.

Διάγραμμα του εννοιολογικό μοντέλου της εφαρμογής, με χρήση της γλώσσας Telos

BEGINTRANSACTION

**TELL** Individual EntityType in M1\_Class end EntityType

**TELL Individual Constrained in M1\_Class end Constrained**

**TELL** Attribute has Attribute from: EntityType to: EntityType in M1\_Class with attribute constrained: Constrained end hasAttribute

**TELL** Individual ProcessType in M1\_Class end ProcessType

**TELL** Attribute input from: EntityType to: ProcessType in M1\_Class end input

**TELL** Attribute control from: EntityType to: ProcessType in M1\_Class end output

**TELL** Attribute mechanism from: EntityType to: ProcessType in M1\_Class end mechanism

**TELL** Attribute output from: ProcessType to: EntityType in M1\_Class end output

**TELL Individual OneToMany in S\_Class, Constrained end OneToMany**

**TELL Individual ZeroToMany in S\_Class, Constrained end ZeroToMany**

**TELL Individual OneToOne in S\_Class, Constrained end OneToOne**

**TELL** Individual FirstName in S\_Class, EntityType end FirstName

**TELL** Individual LastName in S\_Class, EntityType end LastName

**TELL** Individual Patronym in S\_Class, EntityType end Patronym

**TELL** Individual Mothername in S\_Class, EntityType end Mothername

**TELL** Individual Address in S\_Class, EntityType end Address

**TELL** Individual BirthDate in S\_Class, EntityType end BirthDate

**TELL** Individual ADT in S\_Class, EntityType end ADT

**TELL** Individual AFM in S\_Class, EntityType end AFM

**TELL** Individual DOY in S\_Class, EntityType end DOY

**TELL** Individual ID in S\_Class, EntityType end ID

**TELL** Individual Consumer in S\_Class, EntityType end Consumer

**TELL** Individual TransferOrder in S\_Class, ProcessType end TransferOrder

**TELL** Individual ExecutionOrder in S\_Class, ProcessType end ExecutionOrder

**TELL** Attribute mechanism1 from: Consumer to: TransferOrder in S\_Class, mechanism end name

**TELL** Attribute mechanism2 from: Consumer to: ExecutionOrder in S\_Class, mechanism end mechanism2

**TELL** Attribute name from: Consumer to: FirstName in S\_Class, has Attribute with constrained: OneToOne end name

**TELL** Attribute surname from: Consumer to: LastName in S\_Class, has Attribute with constrained: OneToOne end surname

**TELL** Attribute patronym from: Consumer to: Patronym in S\_Class, has Attribute with constrained: OneToOne end patronym

**TELL** Attribute mothernamefrom: Consumerto: Mothernamein S\_Class, hasAttribute end mothername with constrained: OneToOne end mothername

**TELL** Attribute fullAddressfrom: Consumer to: Address in S\_Class, hasAttribute with constrained: OneToMany end fullAddress

**TELL** Attribute fullAddress from: Consumer to: Address in S\_Class, hasAttribute with constrained: OneToMany end fullAddress

**TELL** Attribute birthday from: Consumer to: BirthDate in S\_Class, hasAttribute with constrained: OneToMany end bithday

**TELL** Attribute adt from: Consumer to: ADT in S\_Class, hasAttribute with constrained: OneToMany end adt

**TELL** Attribute afm from: Consumer to: AFM in S\_Class, hasAttribute with constrained: OneToMany end afm

**TELL** Attribute doy from: Consumer to: DOY in S\_Class, hasAttribute with constrained: OneToMany end doy

**TELL** Attribute id from: Consumer to: ID in S\_Class, hasAttribute with constrained: OneToMany end id

**TELL** Individual ConsumerInfo in S\_Class, EntityType end ConsumerInfo

**TELL** Individual TransferOrder in S\_Class, ProcessType end TransferOrder

**TELL** Attribute refersTo from: ConsumerInfo to: Consumer in S\_Class, hasAttribute with constrained: OneToMany end ConsumerInfo

**TELL** Attribute input1 from: ConsumerInfo to: TransferOrder in S\_Class, input end input1

**TELL** Individual ComandPrompt in S\_Class, EntityType end ConsumerInfo

**TELL** Attribute fromConsumer from: ComandPrompt to: Consumer in S\_Class, hasAttribute with constrained: OneToMany end fromConsumer

**TELL** Attribute input2 from: ComandPrompt to: TransferOrder in S\_Class, input end input2

**TELL** Attribute input3 from: ComandPrompt to: ExecutionOrder in S\_Class, input end input3

**TELL** Individual ShareTitle in S\_Class, EntityType end ShareTitle

**TELL** Individual Brand in S\_Class, EntityType end Brand

**TELL** Individual IdNum in S\_Class, EntityType end IdNum

**TELL** Attribute hasBrand from: ShareTitle to: Brand in S\_Class, hasAttribute with constrained: OneToMany end hasBrand

**TELL** Attribute hasIdNum from: ShareTitle to: IdNum in S\_Class, hasAttribute with constrained: OneToMany end hasIdNum

**TELL** Attribute input4 from: ShareTitle to: TransferOrder in S\_Class, input end input4

**TELL** Individual Comand in S\_Class, EntityType end Comand

**TELL** Individual TimeTarget in S\_Class, EntityType end TimeTarget

**TELL** Attribute control5 from: TimeTarget to: ExecutionOrder in S\_Class, control end control5

**TELL** Individual ValueTarget in S\_Class, EntityType end ValueTarget

**TELL** Attribute control3 from: ValueTarget to: ExecutionOrder in S\_Class, control end control3

**TELL** Individual VolumeTarget in S\_Class, EntityType end VolumeTarget

**TELL** Attribute control4 from: VolumeTarget to: ExecutionOrder in S\_Class, control end control4

**TELL** Individual Purchased in S\_Class, EntityType end Purchased

**TELL** Individual Saled in S\_Class, EntityType end Saled

**TELL** Attribute brand from: Comand to: Brand in S\_Class, hasAttribute with constrained: OneToMany end brand

**TELL** Attribute hasTimeSpace from: Comand to: TimeTarget in S\_Class, hasAttribute with constrained: OneToOne end hasTimeSpace

**TELL** Attribute hasValueTarget from: Comand to: ValueTarget in S\_Class, hasAttribute with constrained: OneToOne end hasValueTarget

**TELL** Attribute hasVolumeTarget from: Comand to: VolumeTarget in S\_Class, hasAttribute with constrained: OneToOne end hasVolumeTarget

**TELL** Attribute isPurchased from: Comand to: Purchased in S\_Class, hasAttribute with constrained: OneToOne end isPurchased

**TELL** Attribute isSaled from: Comand to: Saled in S\_Class, hasAttribute with constrained: OneToOne end isSaled

**TELL** Individual CanselationComand in S\_Class, EntityType isA Comand end CanselationComand

**TELL** Attribute orderBy from: CanselationComand to: Consumer in S\_Class, hasAttribute with constrained: OneToOne end orderBy

**TELL** Attribute input5 from: CanselationComand to: ExecutionOrder in S\_Class, input end input5

**TELL** Individual ConsumerRegistered in S\_Class, EntityType isA Consumer end ConsumerRegistered

**TELL** Attribute control1from: ConsumerRegisteredto: TransferOrder in S\_Class, control end control1

**TELL** Attribute is from: ConsumerRegistered to: Consumer in S\_Class, hasAttribute with constrained: OneToOne end is

**TELL** Individual ComandCondition in S\_Class, EntityType isA Comand end ComandCondition

**TELL** Individual Runs in S\_Class, EntityType end Runs

**TELL** Individual Completed in S\_Class, EntityType end Completed

**TELL** Individual Canceled in S\_Class, EntityType end Canceled

**TELL** Attribute runs from: ComandCondition to: Runs in S\_Class, hasAttribute with constrained: ZeroToMany end runs

**TELL** Attribute completed from: ComandCondition to: Completed in S\_Class, hasAttribute with constrained: OneToOne end completed

**TELL** Attribute canceled from: ComandCondition to: Canceled in S\_Class, hasAttribute with constrained: OneToOne end canceled

**TELL** Attribute control2 from: ComandCondition to: ExecutionOrder in S\_Class, control end control2

**TELL** Attribute output1 from: TransferOrder to: ComandCondition in S\_Class, output end output1

**TELL** Attribute output2 from: TransferOrder to: Comand in S\_Class, output end output2

**TELL** Attribute input6 from: Comand to: ExecutionOrder in S\_Class, input end input6

**TELL** Attribute saled from: ShareTitle to: Saled in S\_Class, hasAttribute with constrained: OneToOne end saled

**TELL** Attribute purchased from: ShareTitle to: Purchased in S\_Class, hasAttribute with constrained: OneToOne end purchased

**TELL** Individual OrderClearance in S\_Class, ProcessType end OrderClearance

**TELL** Attribute output3 from: ExecutionOrder to: ShareTitle in S\_Class, output end output3

**TELL** Attribute input7 from: ShareTitle to: OrderClearance in S\_Class, input end input7

**TELL** Individual CertifiedConsumer in S\_Class, EntityType isA Consumer end CertifiedConsumer

**TELL** Attribute output4 from: TransferOrder to: CertifiedConsumer in S\_Class, output end output4

**TELL** Attribute control6 from: CertifiedConsumer to: ExecutionOrder in S\_Class, control end control6

**TELL** Attribute control7 from: CertifiedConsumer to: OrderClearance in S\_Class, control end control7

**TELL** Individual MessageFailure in S\_Class, EntityType end MessageFailure

**TELL** Attribute goes\_to from: MessageFailure to: Consumer in S\_Class, hasAttribute with constrained: OneToOne end goes\_to

**TELL** Attribute output5 from: TransferOrder to: MessageFailure in S\_Class, output end output5

**TELL** Attribute output6 from: ExecutionOrder to: MessageFailure in S\_Class, output end output6

**TELL** Attribute output7 from: OrderClearance to: MessageFailure in S\_Class, output end output7

**TELL** Individual TotalSuply in S\_Class, EntityType end TotalSuply

**TELL** Attribute output8 from: OrderClearance to: TotalSuply in S\_Class, output end output8

**TELL** Individual PortionUpdate in S\_Class, EntityType end PortionUpdate

**TELL** Individual Securities in S\_Class, EntityType end Securities

**TELL** Individual Cash in S\_Class, EntityType end Cash

**TELL** Attribute output9 from: OrderClearance to: PortionUpdatein S\_Class, output end output9

**TELL** Attribute hasSecurities from: PortionUpdate to: Securities in S\_Class, hasAttribute with constrained: OneToOne end hasSecurities

**TELL** Attribute hasCash from: PortionUpdate to: Cash in S\_Class, hasAttribute with constrained: OneToOne end hasCash

**TELL** Attribute update\_to from: PortionUpdate to: Consumer in S\_Class, hasAttribute with constrained: OneToOne end update\_to

**TELL** Individual Report in S\_Class, EntityType end Report

**TELL** Individual AchivedTime in S\_Class, EntityType end AchivedTime

**TELL** Individual AchivedValue in S\_Class, EntityType end AchivedValue

**TELL** Individual TransactionVolume in S\_Class, EntityType end TransactionVolume

**TELL** Individual NumberOfShares in S\_Class, EntityTypeend NumberOfShares

**TELL** Individual StockTitles in S\_Class, EntityType end StockTitles

**TELL** Individual Type in S\_Class, EntityType end Type

**TELL** Attribute output10 from: OrderClearance to: Report in S\_Class, output end output10

**TELL** Attribute sending\_to from: Report to: Consumer in S\_Class, hasAttribute with constrained: OneToOne end sending\_to

**TELL** Attribute brandy from: Report to: Brand in S\_Class, hasAttribute with constrained: OneToOne end brandy

**TELL** Attribute time from: Report to: AchivedTime in S\_Class, hasAttribute with constrained: OneToOne end time

**TELL** Attribute value from: Report to: AchivedValue in S\_Class, hasAttribute with constrained: OneToOne end value

**TELL** Attribute transactionVolume from: Report to: TransactionVolume in S\_Class, hasAttribute end transactionVolume

**TELL** Attribute numberOfShares from: Report to: NumberOfShares in S\_Class, hasAttribute with constrained: OneToOne end numberOfShares

**TELL** Attribute stokeTitles from: Report to: StockTitles in S\_Class, hasAttribute with constrained: OneToOne end stokeTitles

**TELL** Attribute type from: Report to: Type in S\_Class, hasAttribute with constrained: OneToOne end type

ENDTRANSACTION

Δείγμα αποτελεσμάτων της τελος στην κεντρική λειτουργία Process Type την 1η Σύνθετη λειτουργία με όνομα: TransferOrder. Εμφανίζονται οι είσοδοι, οι μηχανισμοί , οι έλεγχοι και οι έξοδοι καθώς και η κληρονομικότητα από την M1-Class – Μεταμοντέλο.

Εικόνα που περιέχει στιγμιότυπο οθόνης

Περιγραφή που δημιουργήθηκε αυτόματα

Το μεταμοντέλο υποστηρίζει την περαιτέρω ανάλυση κάθε Σύνθετης λειτουργίας στις αντίστοιχες λειτουργίες και κάθε λειτουργίας στις αντίστοιχές της πράξεις.

## Περιορισμοί ακεραιότητας και παραγωγικοί κανόνες που απορρέουν από την εκφώνηση:

**Περιορισμοί Ακεραιότητας:**

1. Για να έχω συγκεκριμένο πελάτη πρέπει να υπάρχει πελάτης στο μητρώο πελατών.
2. Ο πελάτης για να προβεί σε οποιαδήποτε συμπλήρωση εντολής πρέπει να εγκριθεί από το σύστημα.
3. Ο Πελάτης για να προβεί σε αγοραπωλησία μετοχών πρέπει να καταβάλει μια προμήθεια στην εταιρία !=0.
4. Μια πράξη για να είναι έγκυρη πρέπει να είναι αγορά ή πώληση.
5. Η αγοραπωλησία πρέπει να αναφαίρετε σε συγκεκριμένο πλήθος μετοχών !=0.
6. Κάθε εντολή αφορά μόνο σε αγορά ή μόνο σε πώληση και σε μετοχές μιας μόνο εταιρίας.
7. H κατάστασης της εντολής μπορεί να έχει μόνο 3 στάδια: ισχύει, ακυρώθηκε, ολοκληρώθηκε.
8. Επιπρόσθετες αγορές ή πωλήσεις στο πλαίσιο της ίδιας εντολής μπορούν να γίνουν εφ’ όσον το χρονικό διάστημα δεν έχει εξαντληθεί επιτευχθεί ο όγκος - στόχος.
9. Για να υπάρξει προμήθεια πρέπει να υπάρξει αγορά ή πώληση.
10. Για να εκτελεστεί η σύνθετη λειτουργία εκτέλεση εντολής είναι απαραίτητο να υπάρχει εγγεγραμμένος πελάτης καθώς και η ύπαρξη της Εντολής που ισχύει ακόμα και δεν έχει ολοκληρωθεί ή δεν έχει ακυρωθεί από τον πελάτη.

**Παραγωγικοί κανόνες:**

1. Η διαβίβαση εντολής είναι παράγωγη κλάση άρα πηγάζουν οι εξής παραγωγικοί κανόνες:

* Πρέπει να υπάρχει πελάτης
* Πρέπει να υπάρχει διεπαφή πελάτη.

1. To μητρώο πελατών προκύπτει από τους πελάτες.
2. Ο υπολογισμός της προμήθειας προκύπτει από τον συνολικό όγκο της συναλλαγής στην αναφορά.

Η προμήθεια που χρεώνει η εταιρία στους πελάτες της υπολογίζεται ως εξής: Για συνολικό πραγματοποιηθέντα όγκο μεταξύ 0 και Ο1, Π1%, για συνολικό πραγματοποιηθέντα όγκο μεταξύ Ο1 και Ο2, Π2%, και για συνολικό πραγματοποιηθέντα όγκο μεγαλύτερο του Ο2, Π3%, όπου Π1>Π2>Π3 (εννοείται 0<Ο1<Ο2<Ο3).

1. σε περίπτωση πώλησης, υπολογισμό του αναλογούντος φόρου υπερτιμήματος, ο οποίος βασίζεται σε σύγκριση της τιμής κτήσης (η οποία ευρίσκεται από τα στοιχεία που τηρούνται στο Κεντρικό Αποθετήριο Αξιών) και της τιμής πώλησης κάθε τίτλου και λαμβάνει υπ’ όψη τους εκάστοτε ισχύοντες συντελεστές φόρου.
2. Ο επιβεβαιωμένος πελάτης προκύπτει από τους πελάτες και την επιβεβαίωσή τους από την διεπαφή χρήστη.
3. Το μητρώο των εταιριών προκύπτει από τις εταιρίες που είναι εγγεγραμμένες στο χρηματιστήριο Αθηνών.
4. Το μήνυμα σφάλματος προκύπτει είτε αν ισχύουν οι απαγορευτικές συνθήκες (ανεσταλμένη διαπραγμάτευση των μετοχών, το ΧΑΑ παρουσιάζει μεγάλη καθυστέρηση) είτε όταν στην λήξη του χρονικού διαστήματος στόχου δεν έχει επιτευχθεί η τιμή στόχος.