



ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ  
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



## ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

### ΤΕΧΝΟΛΟΓΙΑ OLAP ΜΕ ΕΡΓΑΛΕΙΑ MICROSOFT



Του φοιτητή

Δουλάμη Δημήτρη

Αρ. Μητρώου: 05/2827

Επιβλέπων καθηγητής

Κεραμόπουλος Ευκλείδης

## Θεσσαλονίκη 2016

### ΠΡΟΛΟΓΟΣ

Η πτυχιακή εργασία με τίτλο «Τεχνολογία OLAP με εργαλεία Microsoft» έγινε στο πλαίσιο των σπουδών μου στο Τμήμα Πληροφορικής του Αλεξάνδρειου Τεχνολογικού Εκπαιδευτικού Ιδρύματος Θεσσαλονίκης.

Στην πτυχιακή αυτή γίνεται ενδελεχής ανάλυση της τεχνολογίας OLAP χρησιμοποιώντας Microsoft SQL Server, Microsoft SQL Server Analysis Services (SSAS) , Microsoft SQL Server Integration Services (SSIS) και Microsoft Visual Studio.

OLAP (Online Analytical processing) είναι το επόμενο βήμα από το κλασικό data warehousing. Η τεχνολογία αυτή σαν μέρος του BI (Business Intelligence) μιας εταιρείας ή ενός Οργανισμού μπορεί να χρησιμοποιηθεί για να επεξεργαστεί τα δεδομένα που περιέχονται στην βάση για την εξαγωγή συμπερασμάτων και την λήψη αποφάσεων που μπορούν να καθορίσουν το μελλοντικό σχεδιασμό μιας εταιρείας.

## ΠΕΡΙΛΗΨΗ

Η πτυχιακή αυτή παρουσιάζει και αναλύει εις βάθος την τεχνολογία OLAP χρησιμοποιώντας εργαλεία της Microsoft όπως SQL Server και Visual Studio με Business Intelligence και Server Analysis.

Στο πρώτο κεφάλαιο αναλύουμε την βάση όπου στεγάζει τα δεδομένα από τις εφαρμογές και την αποθήκη δεδομένων όπου στεγάζει τα δεδομένα για την αναλυτική επεξεργασία το Business Intelligence.

Στο δεύτερο κεφάλαιο αναλύουμε την τεχνολογία OLAP ανακτώντας τα δεδομένα από την Αποθήκη Δεδομένων και χρησιμοποιώντας τα για να δημιουργήσουμε και να γεμίσουμε τον κύβο. Τα δεδομένα οπτικοποιούνται στο κύβο μέσω διάφορων τεχνικών-operations για την εξαγωγή πληροφορίας.

Στο τρίτο κεφάλαιο παρουσιάζουμε παραδείγματα με την χρήση των εργαλείων της Microsoft και αναλύουμε τις δυνατότητες που μας προσφέρουν, ώστε να έχουμε μία ολοκληρωμένη λύση OLAP. Δημιουργούμε Αποθήκες δεδομένων και διαδικασίες που γεμίζουν με δεδομένα τις αποθήκες. Τέλος δημιουργούμε τους κύβους και τους αναλύουμε, ολοκληρώνοντας με αυτό τον τρόπο την τεχνολογία OLAP.

## ABSTRACT

This thesis presents and analyses OLAP technology in depth using Microsoft tools such as SQL Server and Visual Studio with Business Intelligence Server Analysis.

In the first chapter we analyze the database that host the data from the various applications and the Data Warehouse that hosts the data for the Business Intelligence analytical process

In the second chapter we analyze the OLAP technology by extracting the data from the Data Warehouse and using them to create and fill the cube. The data gets visualized through various techniques – operations to extract information.

In the third chapter we are presenting examples using Microsoft tools and we analyze the capabilities that those tools offer, to have a complete OLAP solution. We are creating Data Warehouses and processes that fill the Data Warehouses with data. Concluding we are creating and analyzing OLAP cubes fulfilling by this way the OLAP technology.

## ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Κεραμόπουλο Ευκλείδη για την ευκαιρία να ασχοληθώ να αναλύσω και να παρουσιάσω την τεχνολογία OLAP , την μητέρα μου για την ευκαιρία που μου έδωσε να σπουδάσω στο Τμήμα Πληροφορικής Θεσσαλονίκης και την υποστήριξη της καθ' όλη την διάρκεια «κυρίως οικονομική» που με βοήθησε να ολοκληρώσω την φοίτηση μου.

Επίσης θα ήθελα να ευχαριστήσω του φίλους μου και την γυναίκα μου για την βοήθεια τους κατά την διάρκεια εκπόνησης της πτυχιακής μου.

## ΠΕΡΙΕΧΟΜΕΝΑ

ΠΡΟΛΟΓΟΣ.....	2
ΠΕΡΙΛΗΨΗ.....	3
ABSTRACT .....	4
ΕΥΧΑΡΙΣΤΙΕΣ .....	5
ΠΕΡΙΕΧΟΜΕΝΑ .....	6
Ευρετήριο εικόνων.....	8
Ευρετήριο πινάκων.....	10
ΕΙΣΑΓΩΓΗ.....	11
ΚΕΦΑΛΑΙΟ 1: ΑΠΟΘΗΚΕΣ ΔΕΔΟΜΕΝΩΝ – DATA WAREHOUSES .....	13
ΕΙΣΑΓΩΓΗ.....	13
1.1 Η Βάση Δεδομένων .....	14
1.2 Η μετάβαση στην τεχνική της Αποθήκης Δεδομένων .....	15
1.3 Ορισμός και ανάλυση της Αποθήκης Δεδομένων .....	16
1.4 Αρχιτεκτονική του συστήματος Αποθήκης Δεδομένων .....	17
1.5 Αρχιτεκτονικές και σχεδίαση Αποθηκών Δεδομένων .....	18
1.5.1 Σχεδίαση Σχήματος Αστέρα – Star Schema.....	19
1.5.2 Σχεδίαση Σχήματος Νιφάδας Χιονιού – Snowflake Schema .....	20
1.5.3 Σχεδίαση Σχήματος Νιφάδας Αστεριού – Star flake Schema.....	20
1.5.4 Σχεδίαση Σχήματος Αστερισμού – Constellation Schema .....	21
1.6 Εισαγωγή των δεδομένων στην Αποθήκη δεδομένων - ETL.....	22
1.7 SSIS - SQL Server Integration Service .....	24
1.8 Μετα-δεδομένα.....	25
1.9 SQL Server Management Studio .....	25
1.10 Σύγκριση OLTP - OLAP.....	27
ΕΠΙΛΟΓΟΣ.....	28
ΚΕΦΑΛΑΙΟ 2 OLAP.....	29
ΕΙΣΑΓΩΓΗ.....	29
2.1 OLAP Cube.....	29
2.2 Λειτουργίες OLAP Cube - Operations .....	31
2.2.1 Roll up .....	31
2.2.2 Drill down .....	31
2.2.3 Slice .....	31

2.2.4 Dice.....	32
2.2.5 Pivot.....	32
2.3 Αρχιτεκτονική SQL Server Analysis Services SSAS .....	32
2.4 Microsoft Business Intelligence Development Studio .....	33
2.5 Αρχιτεκτονικές αποθήκευσης OLAP .....	33
2.5.1 Multidimensional OLAP MOLAP .....	34
2.5.2 Relational OLAP ROLAP .....	34
2.5.3 Hybrid OLAP HOLAP .....	35
2.5.4 Σύγκριση ROLAP-HOLAP-MOLAP.....	35
2.6 FASMI TEST.....	36
2.7 MDX Multidimensional Expressions.....	36
ΕΠΙΛΟΓΟΣ.....	39
ΚΕΦΑΛΑΙΟ 3 Τεχνικά παραδείγματα .....	40
ΕΙΣΑΓΩΓΗ.....	40
3.1 Η βάση δεδομένων Hotel Management .....	40
3.2 Η Αποθήκη Δεδομένων Hotel Management DW .....	40
3.3 Η βάση δεδομένων Adventure Works .....	42
3.4 Η Αποθήκη δεδομένων Adventure Works DW.....	44
3.5 SSIS πακέτο ETL Hotel Management και Customer Integration .....	45
3.6 Ο κύβος OLAP Hotel Management Cube .....	66
3.7 SSAS OLAP Cube Adventure Works .....	90
3.7.1 SSAS Calculations .....	90
3.7.2 SSAS KPIs.....	92
3.7.3 SSAS Actions.....	94
3.7.4 SSAS Storage Options και Partitions .....	97
3.8 Ο κύβος OLAP στο Microsoft Excel.....	102
3.9 Εκπαιδευτική Λίστα με Video Tutorials.....	110
ΕΠΙΛΟΓΟΣ.....	110
ΣΥΜΠΕΡΑΣΜΑΤΑ.....	111
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	112
ΟΔΗΓΟΣ ΧΡΗΣΗΣ ΛΟΓΙΣΜΙΚΟΥ.....	113

## Ευρετήριο εικόνων

Εικόνα 1 "OLAP" .....	11
Εικόνα 2 "Data Warehouse" .....	13
Εικόνα 3 "Hotel Management SQL Database" .....	14
Εικόνα 4 "Αρχιτεκτονική Συστήματος Αποθήκης Δεδομένων" .....	18
Εικόνα 5 "Star schema" .....	19
Εικόνα 6 "Snowflake schema" .....	20
Εικόνα 7 "Star flake schema" .....	21
Εικόνα 8 "Constellation Schema" .....	22
Εικόνα 9 "ETL" .....	23
Εικόνα 10 "SSIS" .....	24
Εικόνα 11 "SQL Server 2014" .....	26
Εικόνα 12 "OLAP Cube" .....	30
Εικόνα 13 "Pivot Operation" .....	32
Εικόνα 14 "Microsoft SSAS" .....	32
Εικόνα 15 "Connect to Analysis Server Cube" .....	37
Εικόνα 16 "MDX New Query" .....	38
Εικόνα 17 "Αποθήκη Δεδομένων Hotel Management DW" .....	41
Εικόνα 18 "βάση δεδομένων properties" .....	43
Εικόνα 19 "βάση Δεδομένων properties permissions" .....	44
Εικόνα 20 "Star Schema Adventure Works DW" .....	45
Εικόνα 21 "SSIS Connection Manager" .....	46
Εικόνα 22 "SSIS Connection Managers" .....	47
Εικόνα 23 "SSIS Design Menu" .....	48
Εικόνα 24 "Παράδειγμα Customers Integration Hotel Management" .....	49
Εικόνα 25 "File System Task" .....	50
Εικόνα 26 "Flat File Source" .....	51
Εικόνα 27 "Flat File Source Editor" .....	52
Εικόνα 28 "Browse του αρχείου Customers.txt" .....	53
Εικόνα 29 "Flat File Columns" .....	54
Εικόνα 30 "Data Conversion and OLE DB Destination" .....	55
Εικόνα 31 "Data Conversion and Data Mapping" .....	56
Εικόνα 32 "OLE DB Destination Connection Manager" .....	57



Εικόνα 33 "OLE DB Destination Mappings" .....	58
Εικόνα 34 "Data Flow" .....	59
Εικόνα 35 "Control Flow Run" .....	60
Εικόνα 36 "Data Flow Run" .....	60
Εικόνα 37 "Control Flow" .....	61
Εικόνα 38 "Execute SQL Task" .....	62
Εικόνα 39 "New SQL Server Agent Job" .....	63
Εικόνα 40 "New Job" .....	64
Εικόνα 41 "New Job Step" .....	65
Εικόνα 42 "SQL Server Agent Job Schedules" .....	66
Εικόνα 43 "Νέο Project Analysis Services" .....	67
Εικόνα 44 "Data Source Wizard" .....	68
Εικόνα 45 "Νέο data source - connection Hotel Management DW" .....	69
Εικόνα 46 "Impersonation information" .....	70
Εικόνα 47 "Data Source view Wizard" .....	71
Εικόνα 48 "Data Source Connection" .....	72
Εικόνα 49 "Επιλογή πινάκων Data Source View" .....	73
Εικόνα 50 "Πίνακες Data source View" .....	74
Εικόνα 51 "Data source View Browse" .....	75
Εικόνα 52 "OLAP Cube Measure" .....	76
Εικόνα 53 "OLAP Cube Dimensions" .....	77
Εικόνα 54 "OLAP Cube Measures and Dimensions" .....	78
Εικόνα 55 "OLAP Cube Design Star Schema" .....	79
Εικόνα 56 "Dimension Create Wizard" .....	80
Εικόνα 57 "Select Creation Method" .....	81
Εικόνα 58 "Time Dimension" .....	82
Εικόνα 59 "Time Dimension Attributes and Hierarchies" .....	83
Εικόνα 60 "Cube Design - Dimensions – Add Cube Dimension" .....	84
Εικόνα 61 "Time Dimension - Measure" .....	85
Εικόνα 62 "Wizard σύνδεσης Dimension - Measure" .....	85
Εικόνα 63 "Add attribute to dimension" .....	86
Εικόνα 64 "Rooms Dimension Δημιουργία Ιεραρχίας" .....	87
Εικόνα 65 "Rooms Dimension Browser" .....	88
Εικόνα 66 "OLAP Cube Process" .....	89

Εικόνα 67 "Calculations – Calculated Members" .....	90
Εικόνα 68 "Calculations – Form View – Script View" .....	91
Εικόνα 69 "KPIs" .....	93
Εικόνα 70 "KPIs Browser View" .....	94
Εικόνα 71 "Visual Studio Cube Actions" .....	95
Εικόνα 72 "Action Additional Properties" .....	96
Εικόνα 73 "Deploying Solution" .....	97
Εικόνα 74 "Dimension Storage Options" .....	98
Εικόνα 75 "Adventure Works Partitions" .....	99
Εικόνα 76 "Internet Sales Partition Source" .....	100
Εικόνα 77 "Internet Sales Partition Query Binding" .....	101
Εικόνα 78 "Partition Storage Settings" .....	102
Εικόνα 79 "Excel σύνδεση στον Analysis Server" .....	103
Εικόνα 80 "Excel Παρουσίαση δεδομένων κύβου – Αναφορά συγκεντρωτικού πίνακα" .....	104
Εικόνα 81 "Excel OLAP" .....	105
Εικόνα 82 "Analyze In Excel" .....	106
Εικόνα 83 "OLAP In Excel" .....	106
Εικόνα 84 "OLAP Adventure Works In Excel" .....	107
Εικόνα 85 "OLAP Cube Excel Query" .....	108
Εικόνα 86 "OLAP Cube Excel Actions" .....	109

## Ευρετήριο πινάκων

Πίνακας 1 "OLTP - OLAP" .....	27
Πίνακας 2 "ROLAP - HOLAP - MOLAP" .....	35

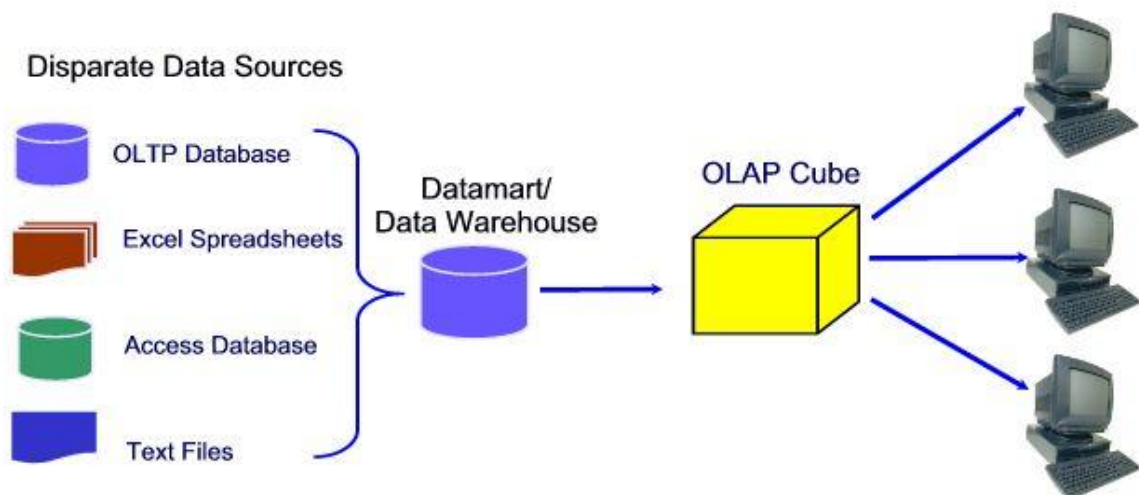
## ΕΙΣΑΓΩΓΗ

Η μηχανογράφηση των διαδικασιών και λειτουργιών έχει μεταβεί από trend στα παλαιότερα χρόνια σε πλήρη αναγκαιότητα για την επιβίωση οποιουδήποτε οργανισμού επιχειρηματικού ή ακαδημαϊκού και γενικότερα. Είναι αδιαμφισβήτητη η εποχή της πληροφόρησης με την ανάγκη για περισσότερη πληροφορία να παραμένει ακόρεστη. Ως αποτέλεσμα τα δεδομένα που συλλέγουν με την πάροδο των ετών στις βάσεις δεδομένων ή σε αρχεία ολοένα και αυξάνονται. Σαν συνέπεια έχουμε τεράστιες βάσεις δεδομένων από πληροφοριακά συστήματα που ολοένα και μεγαλώνουν.

Μια επιχείρηση η οποία συσσωρεύει δεδομένα από τα χρόνια λειτουργίας μπορεί να μελετήσει όλη αυτή την ακατέργαστη πληροφορία που υπάρχει στην βάση και να εξάγει συμπεράσματα που θα την βοηθήσουν να πάρει σημαντικές αποφάσεις στρατηγικής σημασίας για το πώς θα κινηθεί στο μέλλον.

Η λήψη αποφάσεων είναι καθοριστικός παράγοντας για την επιτυχία μιας εταιρείας και τα διοικητικά στελέχη που έχουν επωμιστεί αυτό το καθήκον αναζητούν τα καλύτερα εργαλεία που θα βοηθήσουν στην λήψη αποφάσεων.

Η πτυχιακή αυτή αναλύει πως μπορούν να χρησιμοποιηθούν ορισμένα εργαλεία της Microsoft, μεθοδολογίες και τεχνικές για να αξιοποιήσουμε την βάση δεδομένων που έχουμε στην κατοχή μας και να παράγουμε πληροφορία που θα βοηθήσει στην λήψη αποφάσεων μιας επιχείρησης. Αναλύει σε βάθος την τεχνολογία OLAP (Online Analytical Processing) καθώς και τα εργαλεία που χρειάζονται.



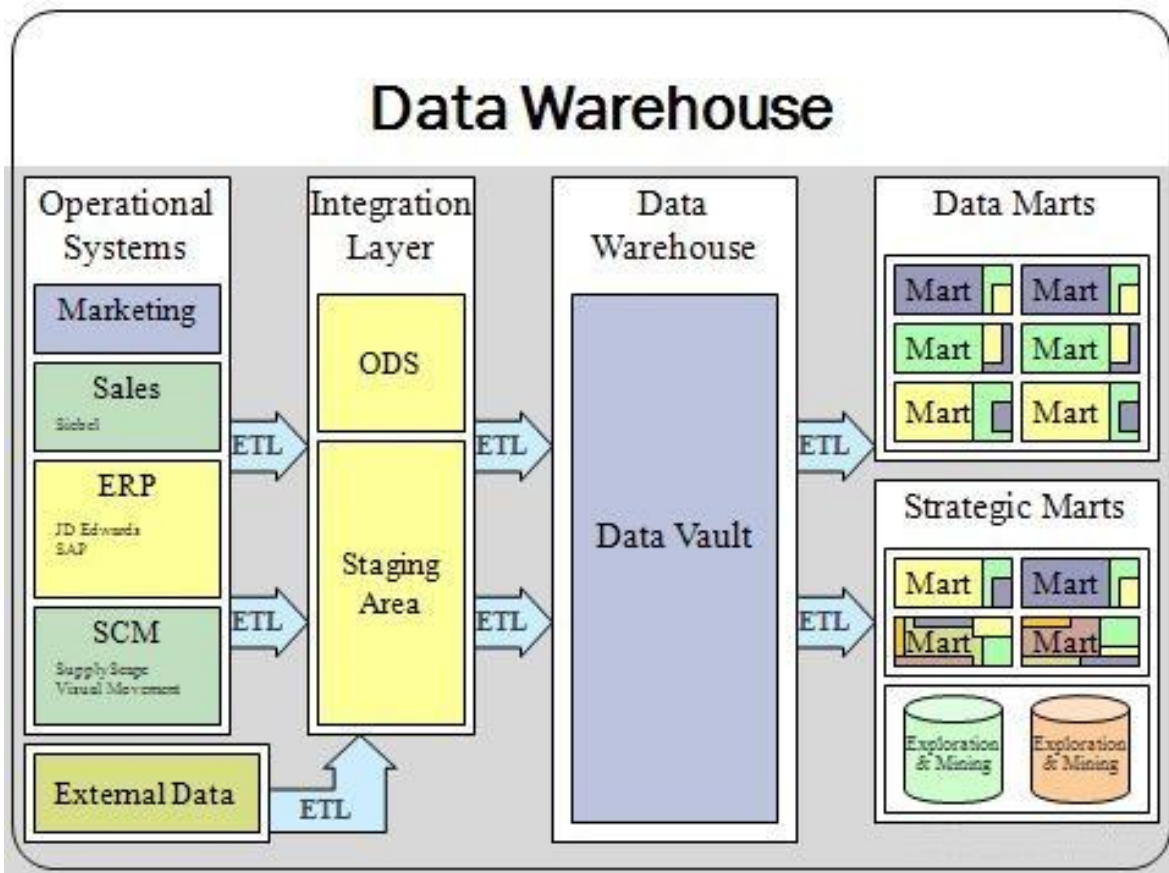
Εικόνα 1 "OLAP"

Στην εικόνα 1 βλέπουμε τις 4 κατηγορίες σε ένα σύστημα OLAP, τα data sources (πηγές δεδομένων), το data warehouse (Αποθήκη δεδομένων), τον OLAP κύβο και τους χρήστες που μελετάνε τον κύβο και τα δεδομένα του. Η εικόνα παρουσιάζει τις πηγές δεδομένων που συσσωρεύουν τα δεδομένα του οργανισμού με την πάροδο των ετών και τροφοδοτούν με την σειρά τους την Αποθήκη Δεδομένων και αυτή με την σειρά της τροφοδοτεί με δεδομένα τον κύβο. Στο τέλος διάφοροι χρήστες όπως developers, αναλυτές, Business Intelligence Officers και Executives με διάφορα εργαλεία χειρίζονται και μελετούν τον κύβο για την εξαγωγή πολύτιμης πληροφορίας.

## ΚΕΦΑΛΑΙΟ 1: ΑΠΟΘΗΚΕΣ ΔΕΔΟΜΕΝΩΝ – DATA WAREHOUSES

### ΕΙΣΑΓΩΓΗ

Βασική απαίτηση ενός συστήματος υποστήριξης αποφάσεων είναι η πρόσβαση στα δεδομένα της επιχείρησης με ένα τρόπο αποδοτικό και εύκολο. Η χρήση της ίδιας της βάσης δεν είναι θεμιτή για λόγους που θα αναλύσουμε στο κεφάλαιο αυτό. Συνεπώς ήταν απαραίτητο να δημιουργηθεί μια ανεξάρτητη «αποθήκη δεδομένων» που θα αντικατοπτρίζει την βάση της επιχείρησης και θα χρησιμοποιηθεί από τα λογισμικά υποστήριξης αποφάσεων.



Εικόνα 2 "Data Warehouse"

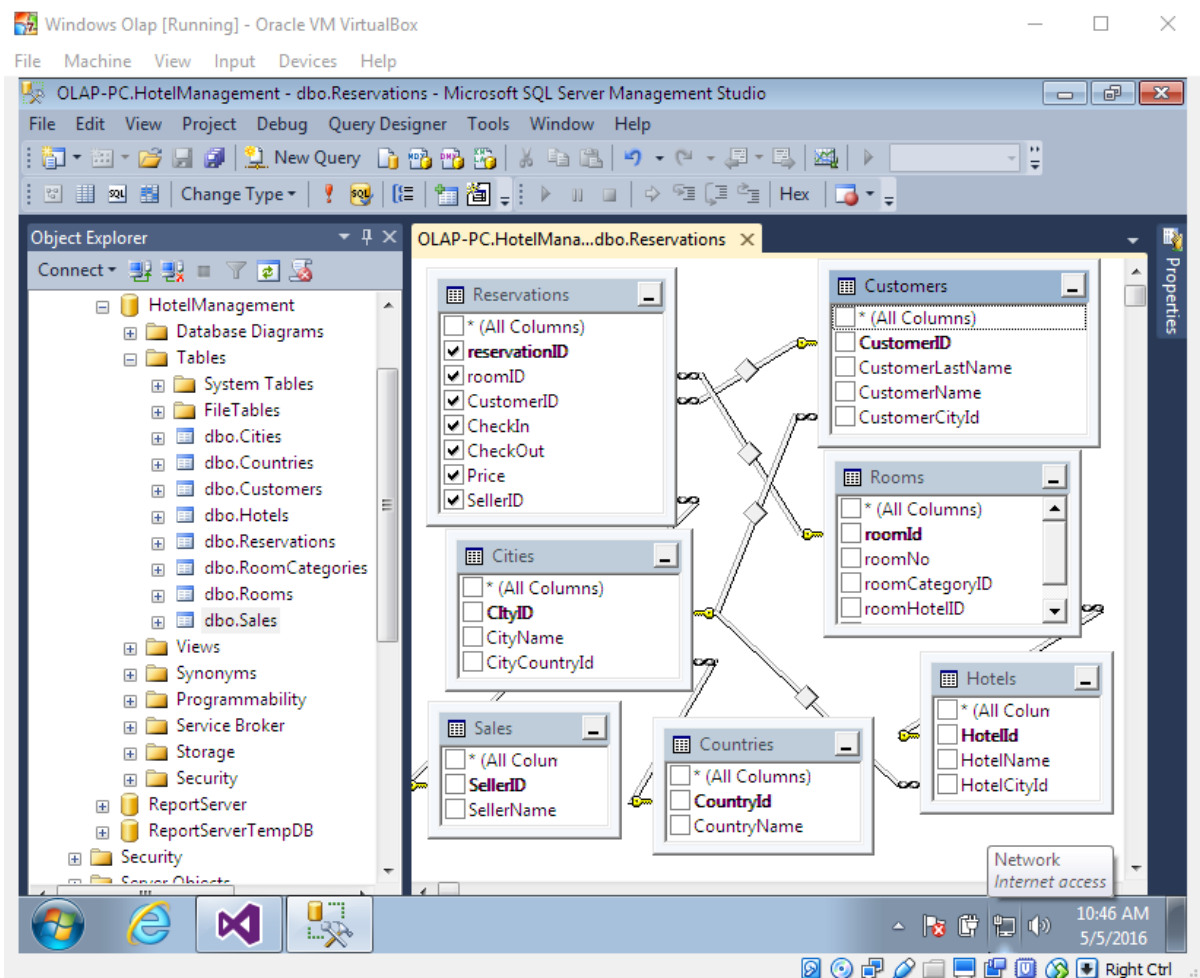
Σε αυτό το κεφάλαιο παρουσιάζονται οι λόγοι που οδήγησαν στην δημιουργία της αποθήκης δεδομένων. Αναλύεται η σχεδίαση και η αρχιτεκτονική τους καθώς και η μεταφορά δεδομένων από την βάση στην Αποθήκη.

Η Microsoft προσφέρει το SQL Management studio για την διαχείριση της βάσης που είναι μέσα στην εγκατάσταση του SQL Server. Για την μεταφορά των δεδομένων από την αποθήκη στην βάση προτείνεται η χρήση του Visual Studio με Integration Services (SSIS) που βρίσκεται μέσα στο πακέτο Business Intelligence.

## 1.1 Η Βάση Δεδομένων

Οι σχεσιακές βάσεις δεδομένων δημιουργήθηκαν στις αρχές του 1970 και έκτοτε ο κλάδος συνεχώς αυξάνεται και πλέον σαν τεχνολογία έχει καθιερωθεί σε μικρούς και μεγάλους οργανισμούς. Τα τελευταία χρόνια σχεδόν όλες οι λειτουργίες μιας επιχείρησης ή ενός οργανισμού είναι μηχανογραφημένες. Η βάση δεδομένων και τα ίδια τα δεδομένα αποτελούν ένα θησαυροφυλάκιο που συνεχώς μεγαλώνει και εξελίσσεται και πρέπει να διατηρηθεί ακέραιο και σε ομαλή λειτουργία με κάθε κόστος. Αυτό οδήγησε τις εταιρείες να φυλάνε τα δεδομένα των βάσεων τους σε διάφορους server στο internet και σε Hard copies μαγνητικών δίσκων με πολλαπλά backup ανά τακτά χρονικά διαστήματα.

Στην πτυχιακή αυτή δημιουργήσαμε στον SQL Server 2014 την βάση δεδομένων «Hotel Management» ενός πληροφοριακού συστήματος κρατήσεων ξενοδοχειακών μονάδων. Η αρχιτεκτονική της βάσης και η ίδια η βάση δημιουργήθηκε από το μηδέν με τελικό σκοπό να προσομοιώσουμε ένα σύστημα μιας επιχείρησης στο οποίο θέλουμε να εφαρμόσουμε OLAP.



Εικόνα 3 "Hotel Management SQL Database"

Η αρχιτεκτονική της βάσης έχει την λογική ότι τα συγκεκριμένα ξενοδοχεία χρησιμοποιούν το ίδιο λογισμικό για την εποπτεία τους και τις κρατήσεις καθώς διαχειρίζονται από την ίδια διοίκηση.

Όπως φαίνεται στην εικόνα 3 ο πίνακας «reservations» αποθηκεύει την κράτηση ενός δωματίου σε ένα ξενοδοχείο με foreign keys από τους πίνακες customer, rooms και sales. Ο πίνακας «sales» έχει τους πωλητές που κάνουν την πώληση μιας κράτησης και μπορεί να είναι είτε μια ιστοσελίδα για κρατήσεις είτε ένα ταξιδιωτικό γραφείο είτε κάποιο ξενοδοχείο. Ο πίνακας «customers» έχει τους πελάτες. Ο πίνακας «rooms» έχει τα δωμάτια των ξενοδοχείων με foreign key to «roomhotelid» , με αυτό τον τρόπο συνδέονται τα δωμάτια με το ξενοδοχείο που ανήκουν. Ο πίνακας «hotels» έχει τα ξενοδοχεία με foreign key to «hotelcityid» που τα συνδέει με την πόλη που βρίσκονται και αντίστοιχα ο πίνακας cities με foreign key to «citycountryID» που συνδέει την πόλη με την χώρα που βρίσκεται.

Ο σκοπός της αρχιτεκτονικής της βάσης είναι να παρέχει transactional processing στα πληροφοριακά συστήματα με τον πλέον αποδοτικό τρόπο.

## 1.2 Η μετάβαση στην τεχνική της Αποθήκης Δεδομένων

Οι λειτουργικές ανάγκες οργανισμών και επιχειρήσεων οδήγησαν στην ανάπτυξη πληροφοριακών συστημάτων με μεγάλες βάσεις δεδομένων. Συστήματα Διαχείρισης Βάσεων Δεδομένων (ΣΔΒΔ) έκαναν την εμφάνιση τους για την ορθή σχεδίαση ανάπτυξη και λειτουργία των Βάσεων Δεδομένων.

Η επεξεργασία των δεδομένων σε αυτές τις βάσεις ονομάστηκε OLTP (Online Transaction processing). Αυτά καλύπτουν καθημερινά την λειτουργία της επιχείρησης και διεκπεραιώνουν συναλλαγές καταχώρησης, διαγραφής ή ανανεώσεις στην Βάση Δεδομένων.

Η βάση της επιχείρησης με την πρώτη ματιά θα ήταν το ιδανικό μέρος για να χρησιμοποιηθεί σαν ένα σύστημα Business Intelligence (BI) και να γίνουν reports.

Το Business Intelligence (BI) είναι ένας αρκετά γενικός όρος που περιλαμβάνει εφαρμογές, εργαλεία, μεθοδολογίες και βέλτιστες πρακτικές για την ανάλυση επιχειρηματικών πληροφοριών - δεδομένων για την λήψη ορθών αποφάσεων, την επίτευξη στόχων και την βελτίωση της απόδοσης σε έναν οργανισμό. Με λίγα λόγια οι τεχνολογίες OLTP και OLAP είναι μέρος του BI μιας επιχείρησης. Το BI έχει εδραιωθεί πλέον στις επιχειρήσεις και αποτελεί βασικό κομμάτι τους.

Τα καθήκοντα του BI σε μια επιχείρηση επωμίζονται σε SQL developers, Business Intelligence Officers, Quality officers, κλπ.

Η Βάση όμως μιας επιχείρησης δημιουργήθηκε αρχικά με τον σκοπό να στεγάσει την πληροφορία των διαφόρων πληροφοριακών συστημάτων της επιχείρησης και να τα υποστηρίξει. Το να χρησιμοποιηθεί για λόγους BI θα μπορούσε να αποβεί μοιραίο λάθος για τους εξής λόγους:

1. Η βάση σχεδιάστηκε αποκλειστικά για τις απαιτήσεις μιας εφαρμογής με στόχο την βέλτιστη απόδοση της εφαρμογής. Αυτό επιτυγχάνετε με απλούστερα και μικρότερα transactions τα οποία δεν επιβαρύνουν σημαντικά την βάση. Τα ερωτήματα στρατηγικής του οργανισμού μπορεί να γίνουν αρκετά πολύπλοκα και χρειάζονται πρόσβαση σε μεγάλο αριθμό δεδομένων που θα έχουν μεγάλο κόστος στην ομαλή λειτουργία του πληροφοριακού συστήματος που χρησιμοποιεί την ίδια βάση.
2. Τα δεδομένα στην βάση είναι ακατέργαστα δεδομένα και δεν γίνεται να παραχθεί αξιόπιστη πληροφορία χωρίς πρώτα να υποστούν επεξεργασία. Μπορεί να υπάρχουν ελλιπή στοιχεία, null εγγραφές, διάφορες ασυνέπειες και η μορφοποίησή τους να τυγχάνει ιδιαίτερης μεταχείρισης.
3. Τα δεδομένα μπορεί να βρίσκονται σε πολλές και διάσπαρτες βάσεις ή σε flat files, comma delimited files και Excel documents με αποτέλεσμα για την ανάλυσή τους να χρειάζεται να ομογενοποιηθούν.
4. Σε αρκετές περιπτώσεις τα δεδομένα στα συστήματα μεταβάλλονται και ανανεώνονται χωρίς να κρατείτε επαρκή ιστορικό για την εξαγωγή χρήσιμων συμπερασμάτων.

Από τα παραπάνω γίνεται σαφές ότι είναι δυσχερής η χρήση των ίδιων των βάσεων των διαφόρων εφαρμογών του οργανισμού και από τα συστήματα στήριξης αποφάσεων. Εδώ γίνεται η εισαγωγή της έννοιας «Αποθήκη Δεδομένων» η οποία δίνει λύση στα παραπάνω προβλήματα. Ιδιαίτερα οι εφαρμογές OLAP επιταχύνονται όταν τα δεδομένα οργανώνονται σε μια αποθήκη δεδομένων με τρόπους μη παραδοσιακούς.

### 1.3 Ορισμός και ανάλυση της Αποθήκης Δεδομένων

Με τον όρο Αποθήκη Δεδομένων χαρακτηρίζουμε ένα σύνολο τεχνολογιών που στεγάζει τα δεδομένα των πληροφοριακών συστημάτων ενός οργανισμού και επιτρέπει στους Business Intelligence Officers και στους Data Analysts να έχουν αποδοτική πρόσβαση σε αυτά.

Η Αποθήκη διατηρεί τα δεδομένα που αντλεί από τις βάσεις του οργανισμού αλλά και δεδομένα που προέρχονται και από εξωτερικές πηγές. Αυτά τα δεδομένα



οργανώνονται κατάλληλα για να απαντήσουν στις κρίσιμες απαιτήσεις των αναλυτών των συστημάτων υποστήριξης αποφάσεων.

Οι βάσεις δεδομένων παρέχουν δυνατότητα για OLTP Συνεχή Επεξεργασία Συναλλαγών ενώ οι αποθήκες δεδομένων παρέχουν δυνατότητα για OLAP Συνεχή Αναλυτική Επεξεργασία των δεδομένων και περιέχουν ιστορικά και συγκεντρωτικά δεδομένα τα οποία είναι πολύ χρήσιμα στην στήριξη αποφάσεων. Σαν μοναδικό στόχο έχουν την αποδοτική απάντηση πολύπλοκων ερωτήσεων κατά την αναλυτική επεξεργασία δεδομένων σε υψηλές ταχύτητες.

Η δημιουργία και η συντήρηση της Αποθήκης είναι μια πολύπλοκη διαδικασία και μπορεί να έχει διαφορετικές προσεγγίσεις. Το σύνηθες είναι να δημιουργηθεί μια αποθήκη που θα περιέχει δεδομένα από όλες τις δραστηριότητες της επιχείρησης με τρόπο αναλυτικό. Μια άλλη πρακτική είναι να δημιουργηθούν επιμέρους συλλογές δεδομένων τα λεγόμενα data marts με κριτήριο το αντικείμενο από το οποίο προέρχονται ή το τμήμα το οποίο αφορούν με σκοπό την περαιτέρω κατηγοριοποίηση και ευελιξία. Στην δεύτερη πρακτική ο κίνδυνος είναι μελλοντικά τα δεδομένα να μην μπορούν να συνδυαστούν και να παρέχουν απαντήσεις σε κοινά προβλήματα.

Από αρκετές εταιρείες για λόγους χρόνου και συντομίας συνηθίζεται να γίνεται Business Intelligence πάνω στην ίδια την βάση δεδομένων της εταιρείας δημιουργώντας reports για την υποστήριξη αποφάσεων. Τα τελευταία χρόνια όλο και περισσότεροι οργανισμοί υιοθετούν τη σχεδίαση και ανάπτυξη Αποθηκών Δεδομένων. Τεράστια ποσά επενδύονται στο Business Intelligence και τα συστήματα αυτά κρίνονται ήδη ως αναγκαία για το μέλλον μιας επιχείρησης. Μεγάλες εταιρείες στο χώρο των Βάσεων Δεδομένων αναπτύσσουν και προτείνουν λογισμικά για την υποστήριξη του τομέα αυτού. Η Microsoft προσφέρει μια πλήρη γκάμα εργαλείων που καλύπτουν όλο το φάσμα του OLAP. Σαν host της βάσης και της Αποθήκης Δεδομένων έχει τον SQL Server και η διαχείριση της βάσης και της Αποθήκης γίνεται μέσω του SQL Management Studio.

#### 1.4 Αρχιτεκτονική του συστήματος Αποθήκης Δεδομένων

Η αρχιτεκτονική μιας Αποθήκης Δεδομένων πρέπει να ικανοποιεί στόχους όπως η αποδοτικότητα του συστήματος, η διαθεσιμότητα του συστήματος και η ενημέρωση του συστήματος.

Τα κύρια δομικά μέρη ενός ολοκληρωμένου συστήματος Αποθήκης Δεδομένων είναι τα εξής:

**Πηγές :** Οι πηγές από όπου αντλούνται δεδομένα που γεμίζουν την Αποθήκη Δεδομένων. Μπορούν να είναι αρχεία ή βάσεις δεδομένων. Στην παρούσα πτυχιακή πηγή είναι ο SQL Server της Microsoft που στεγάζεται η βάση.

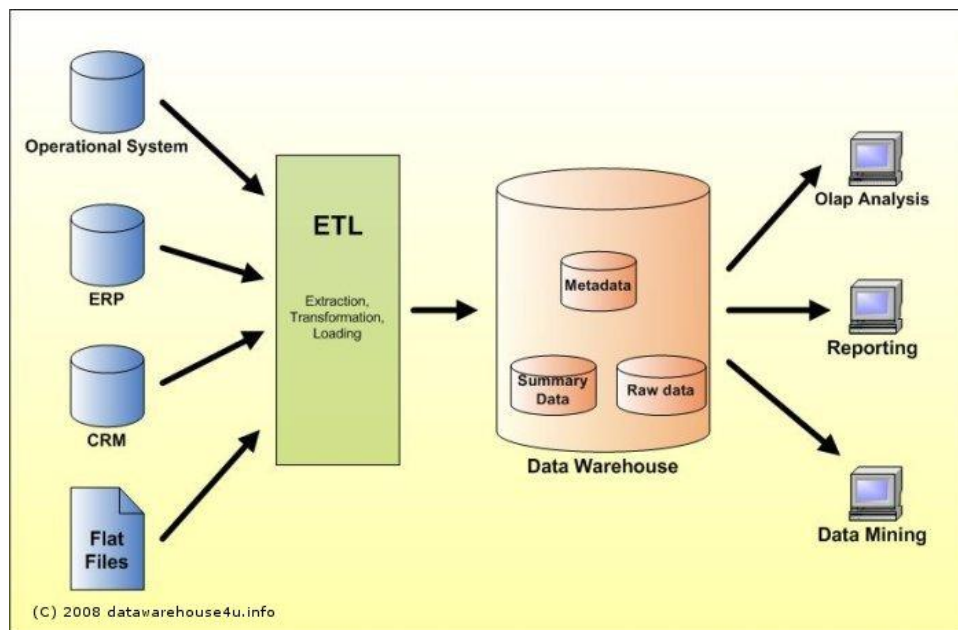
**Μεταφορείς – Μετατροπείς :** Είναι εφαρμογές που εκτελούν τις διαδικασίες μεταφοράς και μορφοποίησης των δεδομένων από τις πηγές στις Αποθήκες Δεδομένων. Στην παρούσα πτυχιακή είναι το SSIS (SQL Server Integration Service) πακέτο το οποίο αναλαμβάνει όλη την διαδικασία ETL (Extract Transform Load)

**Αποθήκη Δεδομένων :** Το σύστημα που αποθηκεύονται τα δεδομένα που θα χρησιμοποιηθούν από τους αναλυτές.

**Βάση Μετα-Δεδομένων :** Είναι το σύστημα που αποθηκεύει πληροφορίες σχετικές με τη δομή και την λειτουργία ολόκληρου του συστήματος.

**Διαχειριστής :** Εφαρμογή διαχείρισης του συστήματος.

**Εφαρμογές Ανάλυσης (OLAP) :** Οι εφαρμογές ως συστήματα στήριξης αποφάσεων που χρησιμοποιούν τα δεδομένα της Αποθήκης Δεδομένων για να εξάγουν πολύτιμη γνώση για τον οργανισμό.



Εικόνα 4 "Αρχιτεκτονική Συστήματος Αποθήκης Δεδομένων"

Στην εικόνα 4 το ETL εξάγει, μετασχηματίζει και φορτώνει τα δεδομένα από τις διάφορες πηγές στην Αποθήκη.

## 1.5 Αρχιτεκτονικές και σχεδίαση Αποθηκών Δεδομένων

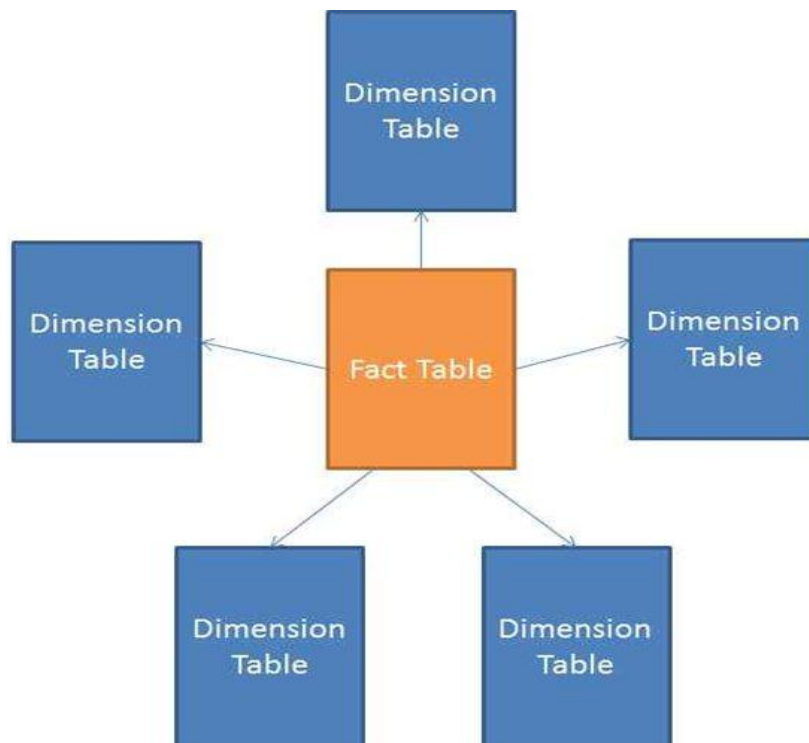
Οι Αποθήκες Δεδομένων χρησιμοποιούνται αποκλειστικά από τα συστήματα υποστήριξης αποφάσεων συνεπώς η σχεδίαση τους είναι διαφορετική από αυτή των κλασικών βάσεων δεδομένων. Τα διαγράμματα Οντοτήτων – Συσχετίσεων και

η κανονικοποίηση που χρησιμοποιούνται στο σχεδιασμό βάσεων δεδομένων κρίνονται ανεπαρκή.

Οι πλέον κατάλληλες τεχνικές για την σχεδίαση μιας Αποθήκης Δεδομένων είναι σχήματος αστέρα, σχήματος νιφάδας χιονιού και σχήμα αστερισμού.

### 1.5.1 Σχεδίαση Σχήματος Αστέρα – Star Schema

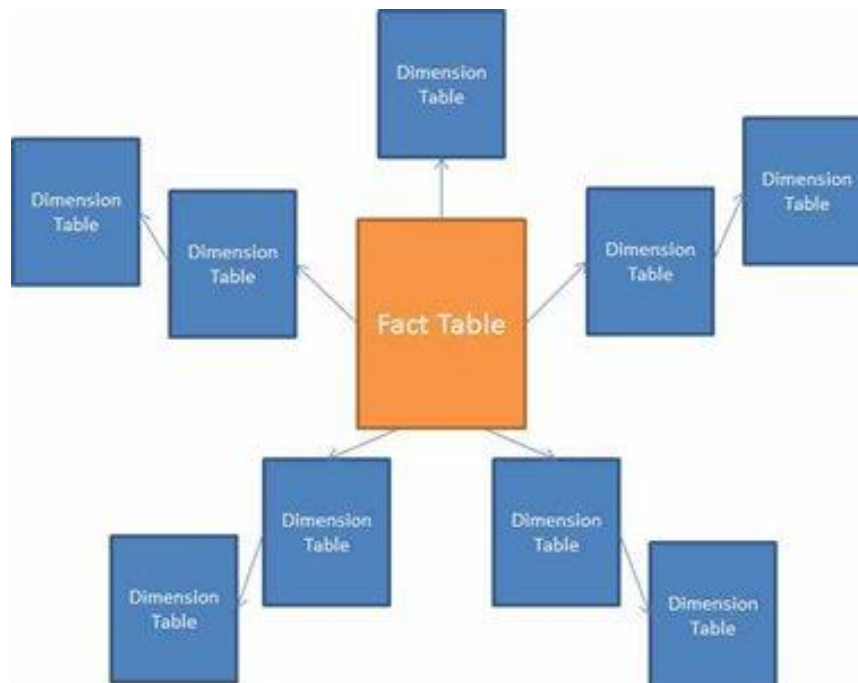
Το σχήμα αστέρα (star schema) είναι μια δομή που περιέχει ένα πίνακα γεγονότων - συμβάντων στο κέντρο ο οποίος είναι περιτριγυρισμένος από πίνακες διαστάσεων. Ο πίνακας γεγονότων (fact table) περιέχει ένα foreign key για κάθε πίνακα διαστάσεων (dimension table). Με αυτό τον τρόπο συνδέονται τα γεγονότα με τις διαστάσεις τους. Επίσης περιέχει μέτρα (measures) τα οποία περιέχουν αριθμητικά δεδομένα. Οι πίνακες διαστάσεων περιέχουν δεδομένα ανάλογα της διάστασης. Ο πίνακας γεγονότων μπορεί να είναι κανονικοποιημένος σε αντίθεση με τους πίνακες διαστάσεων. Τα γεγονότα είναι αμετάβλητα στο χρόνο και οι διαστάσεις παρέχουν περισσότερη πληροφορία για το γεγονός-συμβάν.



Εικόνα 5 "Star schema"

### 1.5.2 Σχεδίαση Σχήματος Νιφάδας Χιονιού – Snowflake Schema

Το σχήμα νιφάδας χιονιού (snowflake schema) είναι μια παραλλαγή του σχήματος αστέρα στο οποίο κάθε διάσταση μπορεί να διαθέτει τις δικές της διαστάσεις. Η ιεραρχία των διαστάσεων στο σχήμα νιφάδας αναπαρίσταται κανονικοποιώντας τους πίνακες διαστάσεων. Το κύριο πλεονέκτημα αυτού του σχήματος είναι ότι αποφεύγεται όσο δυνατόν περισσότερο ο πλεονασμός των δεδομένων. Το μειονέκτημα είναι ότι μεγαλώνει η πολυπλοκότητα των ερωτημάτων για απλά ερωτήματα.

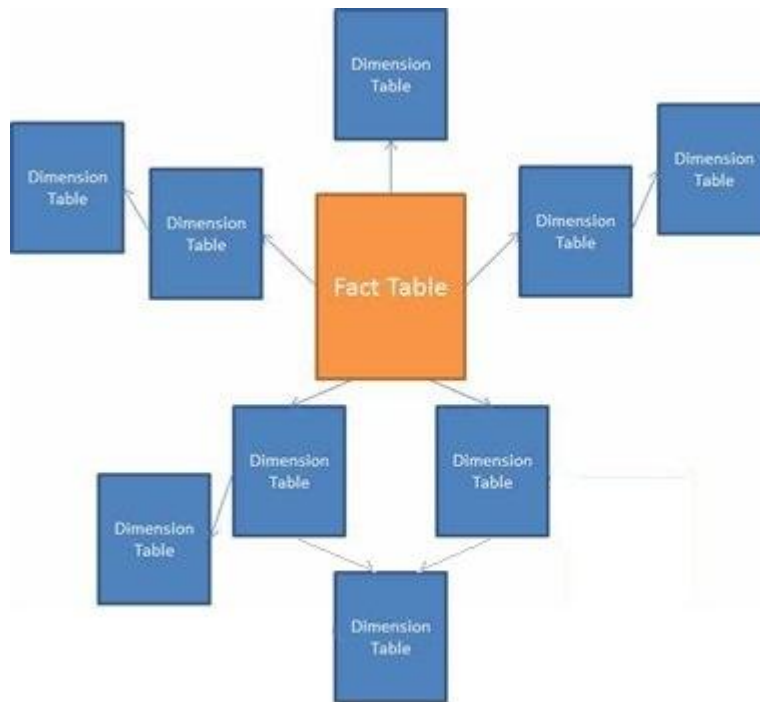


Εικόνα 6 "Snowflake schema"

Στην εικόνα 6 βλέπουμε ότι οι διαστάσεις γίνονται referenced από άλλες διαστάσεις, η σύνδεση αυτή επιτυγχάνεται με την χρήση primary και foreign key ως σημείο αναφοράς μεταξύ των ιδιοτήτων των διαστάσεων.

### 1.5.3 Σχεδίαση Σχήματος Νιφάδας Αστεριού – Star flake Schema

Το σχήμα Νιφάδας Αστεριού (star flake schema) είναι μια υβριδική δομή που περιέχει αποκανονικοποιημένα σχήματα αστεριού και κανονικοποιημένα σχήματα νιφάδας χιονιού.

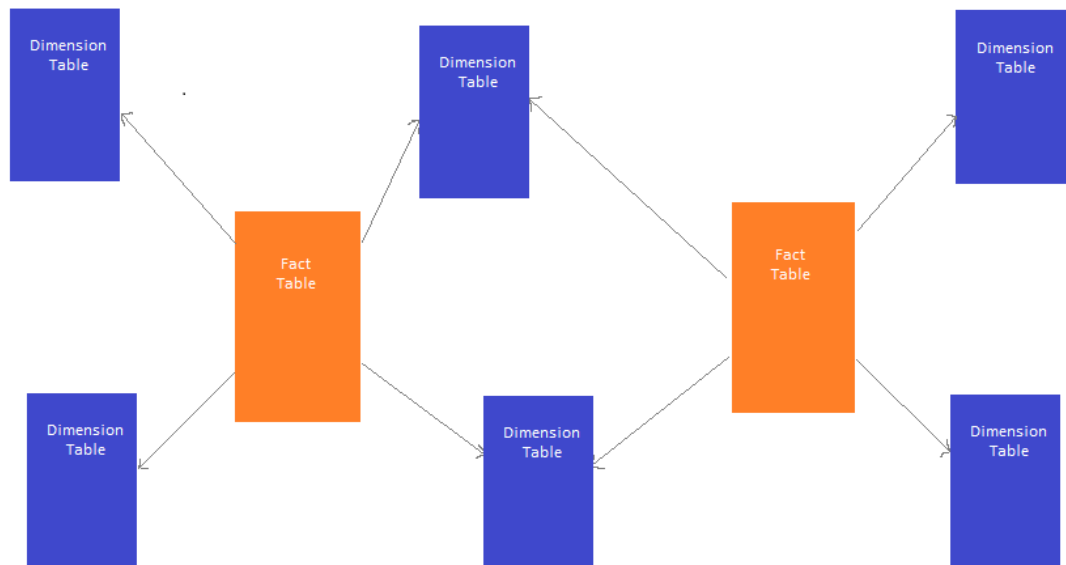


Εικόνα 7 "Star flake schema"

Στην εικόνα 7 βλέπουμε ένα Star Flake Schema στο οποίο κάποιοι πίνακες διαστάσεων μπορεί να είναι normalized και κάποιοι άλλοι πίνακες διαστάσεων να είναι denormalized. Επίσης στην εικόνα αυτή βλέπουμε την έννοια του Outrigger. Το Outrigger είναι η διάσταση που είναι συνδεδεμένη με παραπάνω από μία άλλες διαστάσεις. Με αυτό τον τρόπο η διάσταση γίνεται referenced σε μία άλλη διάσταση. Το primary key του Outrigger γίνεται referenced από το Foreign key των άλλων διαστάσεων.

#### 1.5.4 Σχεδίαση Σχήματος Αστερισμού – Constellation Schema

Το σχήμα Αστερισμού (constellation schema) είναι μια παραλλαγή του σχήματος αστέρα για πιο σύνθετες εφαρμογές. Στο σχήμα αυτό υπάρχουν περισσότεροι από έναν κεντρικοί πίνακες γεγονότων. Η σχεδίαση του μπορεί να είναι αρκετά πολύπλοκη.



Εικόνα 8 "Constellation Schema"

Το σχήμα αστερισμού συναντάται σε πιο σύνθετες εφαρμογές και σε πολύ-θεματικές αποθήκες δεδομένων.

### 1.6 Εισαγωγή των δεδομένων στην Αποθήκη δεδομένων - ETL

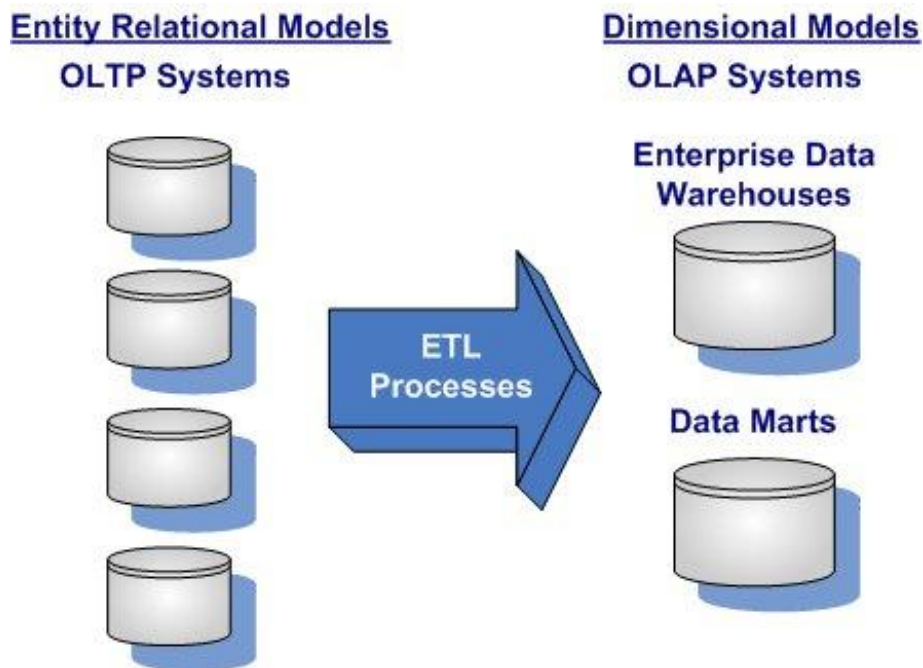
Τα δεδομένα εξάγονται από τις πηγές, καθαρίζονται, μορφοποιούνται-μετασχηματίζονται και στο τέλος φορτώνονται στην αποθήκη δεδομένων. Η διαδικασία ονομάζεται ETL (Extract Transform Load).

Κατά το extract τα δεδομένα εξορύσσονται από τις διάφορες πηγές δεδομένων. Οι πηγές μπορεί να είναι :

- Βάσεις δεδομένων της εταιρείας.
- Εξωτερικά δεδομένα όπως οικονομικοί δείκτες, ο καιρός και διάφορα ανεξάρτητα δεδομένα που δεν σχετίζονται άμεσα με την επιχείρηση αλλά μπορούν να συνδυαστούν για πρόσθετη αξία.
- Αρχεία δεδομένων όπως coma delimited files, Excel, CSV.

Για κάθε πηγή θα χρειαστεί ένα σύστημα που θα ολοκληρώσει την διαδικασία από την εξαγωγή μέχρι την φόρτωση στην αποθήκη. Στην περίπτωση μας που η πηγή είναι βάση δεδομένων σε SQL Server θα χρειαστεί να φτιαχτεί ένα SSIS (SQL Server Integration Service) πακέτο το οποίο θα αυτοματοποιήσει την διαδικασία ETL. Το SSIS χρησιμοποιείται κατά κόρον για επαναλαμβανόμενες διαδικασίες

πάνω στον SQL Server, Θα εξάγει τα δεδομένα και το επόμενο βήμα είναι ο καθαρισμός και η μορφοποίηση όπου αυτό κριθεί απαραίτητο. Ο καθαρισμός περιλαμβάνει διόρθωση ασυνεπειών ή απομάκρυνση κενών εγγραφών. Η μορφοποίηση περιλαμβάνει την διαμόρφωση των δεδομένων με σκοπό να εξυπηρετηθεί το βάθος της λεπτομέρειας που επιζητείται και η αποδοτικότητα του συστήματος. Επίσης σημαντική είναι η χρονοσήμανση των δεδομένων για λόγους ιστορικότητας. Στο τέλος τα καθαρισμένα και μετασχηματισμένα δεδομένα εισάγονται στους καινούριους πίνακες της Αποθήκης.

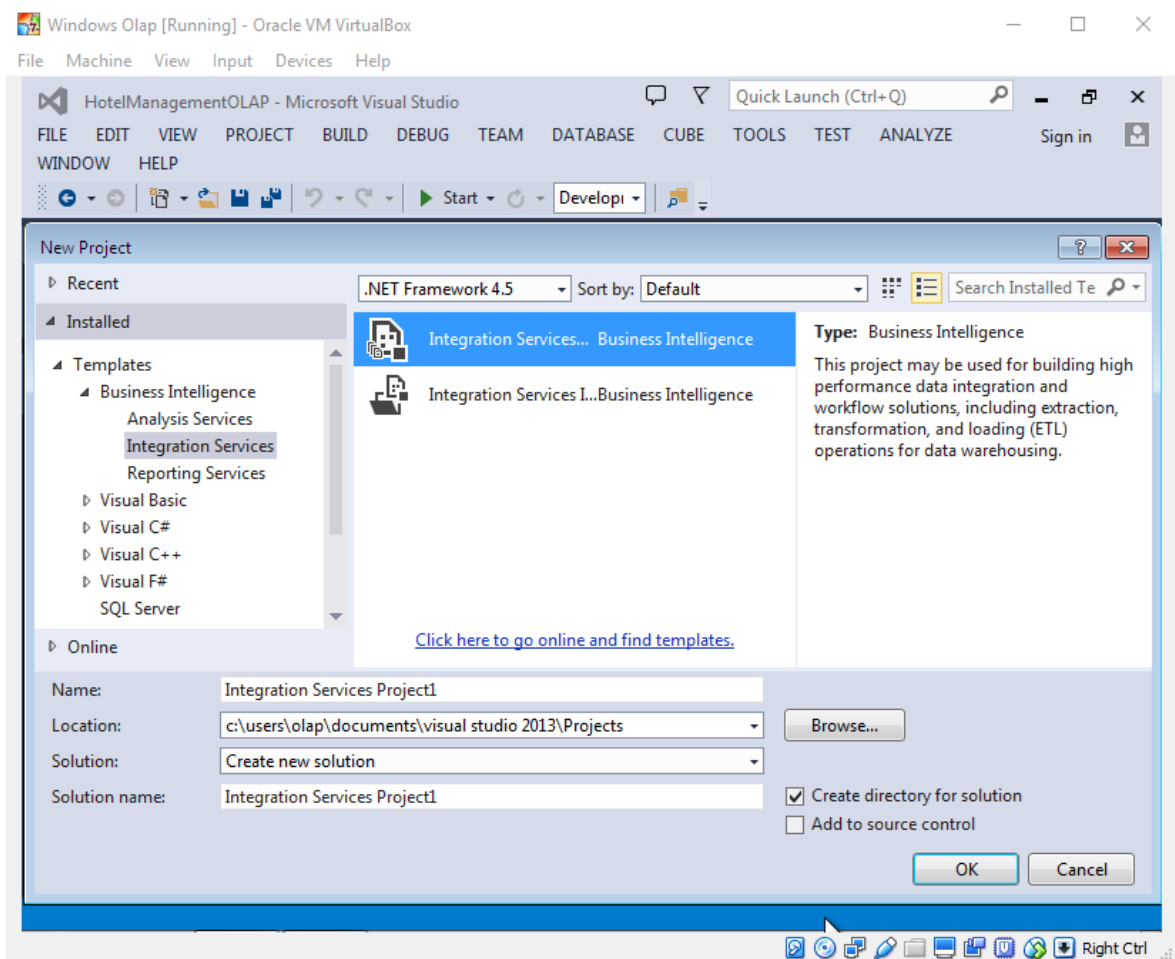


Εικόνα 9 "ETL"

Η διαδικασία ETL έχει μεγάλο υπολογιστικό κόστος τόσο στις πηγές όσο και στην Αποθήκη Δεδομένων επομένως αυτή η διαδικασία πρέπει να γίνεται μαζικά σε περιοδικά χρονικά διαστήματα και πρέπει να ελαχιστοποιείται ο φόρτος στο σύστημα. Σε μια επιχείρηση που δουλεύει 24 ώρες το 24ωρο 7 ημέρες την εβδομάδα το SSIS που θα χειριστεί το ETL για να μεταφέρει τα δεδομένα του προηγούμενου χρόνου στην Αποθήκη δεδομένων θα μπορούσε να επιβαρύνει σημαντικά το σύστημα και τους server με ότι αυτό συνεπάγεται για τις εφαρμογές και τα πληροφοριακά συστήματα της επιχείρησης.

## 1.7 SSIS - SQL Server Integration Service

Το SSIS είναι μια πλατφόρμα για την δημιουργία υψηλής απόδοσης ETL πακέτων. Η πλατφόρμα αυτή είναι ενσωματωμένη στο Visual Studio με Integration Services και Business Intelligence. Αποτελεί ένα γραφικό περιβάλλον με πολλές δυνατότητες διαχείρισης και επεξεργασίας δεδομένων από και προς την βάση δεδομένων.



Εικόνα 10 "SSIS"

Το SSIS παρέχει δυνατότητες αντιγραφής, μεταφοράς, επεξεργασίας, διαγραφής και εισαγωγής δεδομένων ενός SQL πίνακα σε ένα άλλο καθώς και backup – restore. Επίσης έχει δυνατότητες εξαγωγής και επεξεργασίας δεδομένων από αρχεία, όπως comma delimited files, excel, CSV κλπ. Υποστηρίζει SQL scripts και παρέχει δυνατότητα scheduling στην οποία μπορεί να τρέχει αυτόματα το πακέτο SSIS ανά τακτά χρονικά διαστήματα ή στις ορισμένες ημερομηνίες επιτυγχάνοντας πλήρη αυτοματοποίηση.

Η ορθή πρακτική σε ένα οργανισμό σχεδιάζοντας το ETL είναι να απασχολήσουμε όσο το δυνατόν λιγότερο την βάση δεδομένων και την αποθήκη δεδομένων.



Η βάση χρησιμοποιείται από τις εφαρμογές του οργανισμού και η αποθήκη από τον OLAP κύβο και τα reports. Συνεπώς δεν θέλουμε το ETL να δεσμεύει πολλά resources από την βάση και από την αποθήκη. Ο τρόπος να το επιτύχουμε αυτό είναι να δημιουργήσουμε μια intermediate βάση πανομοιότυπη της αρχικής στην οποία θα παίρνουμε copy κάθε φορά τα δεδομένα για την εισαγωγή στην αποθήκη έτσι θα απασχολούμε όσο το δυνατόν λιγότερο την κύρια βάση δεδομένων. Αφού μεταφέρουμε τα δεδομένα στην intermediate βάση έχουμε αποδεσμεύσει την κύρια βάση θα μπορούμε να κάνουμε την οποιαδήποτε επεξεργασία χρειαζόμαστε πάνω τους. Τέλος θα τα μεταφέρουμε τροποποιημένα στην Αποθήκη Δεδομένων.

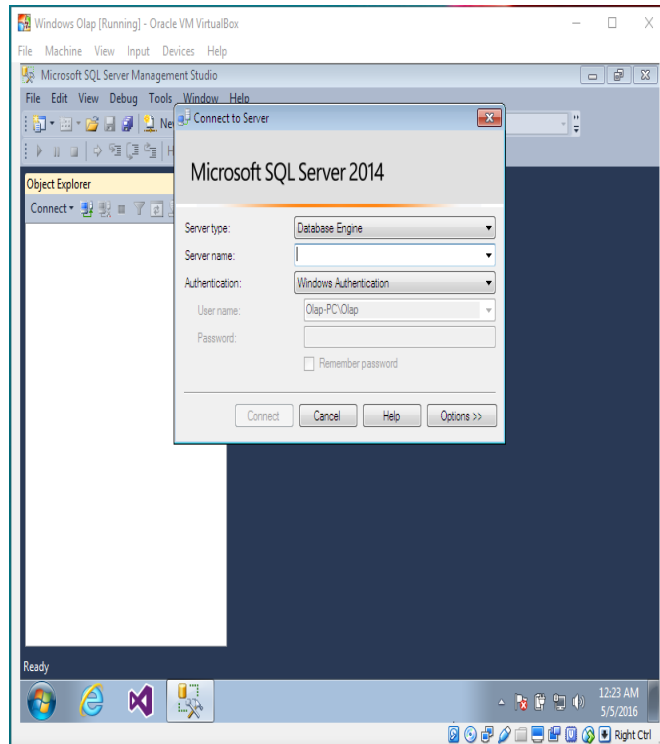
## 1.8 Μετα-δεδομένα

Τα μετα-δεδομένα είναι δεδομένα σχετικά με τα αρχικά δεδομένα της Αποθήκης. Ο σκοπός τους είναι η καλύτερη κατανόηση των δεδομένων της αποθήκης. Μπορεί να υπάρχουν μετα-δεδομένα που περιγράφουν τα δεδομένα όπως τους τύπους τους, τις σχέσεις μεταξύ τους και τους κανόνες αναπαράστασης. Το σύστημα υποστήριξης αποφάσεων επίσης μετά την ανάλυση παράγει κάποια δεδομένα μέσα από τις διαδικασίες του τα οποία μπορούν να αποθηκευτούν με την μορφή μετα-δεδομένων. Τα μετα-δεδομένα εν συντομία περιέχουν :

- Λεξικό Δεδομένων για τον ορισμό και την περιγραφή των δεδομένων που αποθηκεύονται στην αποθήκη και τις μεταξύ τους συσχετίσεις.
- Περιγραφή της ροής δεδομένων στο σύστημα.
- Περιγραφή των κανόνων μετατροπής κατά την μεταφορά.
- Δεδομένα ελέγχου versioning των δεδομένων.
- Στατιστικά χρήσης.
- Κανόνες και πληροφορία σχετικά με την πρόσβαση στην Αποθήκη.
- Ονόματα και aliases.

## 1.9 SQL Server Management Studio

Ο SQL Server είναι ο server που κάνει hosting την βάση δεδομένων καθώς και την αποθήκη δεδομένων στην παρούσα πτυχιακή. Ο SQL Server Management Studio είναι το διαχειριστικό περιβάλλον για τις σχεσιακές βάσεις δεδομένων τις αποθήκες δεδομένων κάποια Reporting Services (SSRS) και Integration Services (SSIS). Στην πτυχιακή αυτή χρησιμοποιούμε τον Microsoft SQL Server 2014 για να δημιουργήσουμε την βάση δεδομένων «Hotel Management» που θα περιέχει τα δεδομένα ενός πληροφοριακού συστήματος διαχείρισης ξενοδοχείων.



Εικόνα 11 "SQL Server 2014"

Το Integration Service μπορεί να χρησιμοποιηθεί για να γίνει το ETL από την βάση στην αποθήκη δεδομένων ώστε να δημιουργηθεί μια αυτοματοποιημένη διαδικασία που θα τρέχει και θα ενημερώνει την αποθήκη με τα καινούρια δεδομένων της βάσης.

- **Processing analysis Services Objects** Το Management Studio παρέχει δυνατότητες επεξεργασίας του κύβου γεμίζοντας τους με δεδομένα από την αποθήκη δεδομένων.
- **Browsing analysis services objects** Έχει δυνατότητα browsing του κύβου με drag and drop τεχνικές για browsing των ιδιοτήτων, των ιεραρχιών και των μετρικών σε πάνελ για γρήγορη οπτικοποίηση των δεδομένων.
- **Queries** Μπορούν να γραφούν multidimensional queries (MDX) όπου είναι και η γλώσσα προγραμματισμού πολυδιάστατων κύβων.
- **Scripting** Επίσης μπορούν να γραφούν scripts με την μορφή XMLA για την δημιουργία αντικειμένων.
- **Management** Τέλος υποστηρίζει θέματα ασφάλειας των δεδομένων με ιεραρχίες security και administration καθώς και backup - restore.

## 1.10 Σύγκριση OLTP - OLAP

Το OLTP Online Transaction Processing χαρακτηρίζεται από μεγάλο αριθμό transactions όπως insert, update και delete. Transactions που γίνονται γρήγορα και απλά, με απλής λογικής και χαμηλού φόρτου queries, με σκοπό την καταγραφή δεδομένων για την όσο δυνατόν καλύτερη αποδοτικότητα των πληροφοριακών συστημάτων. Οι πίνακες και τα δεδομένα είναι κανονικοποιημένα συνήθως επιπέδου 3NF. Τα δεδομένα ενημερώνονται και ενδέχεται να αλλάζουν συνεχώς.

Το OLAP online analytical Processing χαρακτηρίζεται από χαμηλό αριθμό από transactions. Τα queries είναι πολύπλοκα και υψηλού φόρτου. Χρησιμοποιείται για λόγους εξόρυξης πληροφορίας από τα δεδομένα. Τα δεδομένα είναι συνήθως αποκανονικοποιημένα και ενοποιημένα. Τα δεδομένα μιας αποθήκης είναι ιστορικά και time stamped και δεν υπόκεινται σε αλλαγές.

Πίνακας 1 "OLTP - OLAP"

	<b>OLTP</b>	<b>OLAP</b>
Σκοπός Δεδομένων	Βασικές επιχειρησιακές λειτουργίες	Υποστήριξη Αποφάσεων, Στρατηγικές
Πηγές	Λειτουργικά Δεδομένα από OLTP συστήματα	Ενοποιημένα Δεδομένα από OLTP συστήματα
Queries	Απλά queries για σχετικά λίγες εγγραφές	Πολύπλοκα queries με συναθροίσεις πινάκων
Ταχύτητα επεξεργασίας	Γρήγορη επεξεργασία	Αργή επεξεργασία λόγω τεράστιου όγκου δεδομένων
Μέγεθος	Gigabytes Τα ιστορικά δεδομένα συνήθως γίνονται archived	Terabytes Χρήση ιστορικών δεδομένων
Backup and Recovery	Συχνό backup Κρίσιμα δεδομένα για την λειτουργία της επιχείρησης	Τα δεδομένα ξαναφορτώνονται από τα OLTP συστήματα
Σχεδίαση	Κανονικοποιημένα με πολλούς πίνακες	Αποκανονικοποιημένα με λίγους πίνακες

Στον πίνακα 1 συνοψίζονται οι βασικές διαφορές μεταξύ των τεχνολογιών OLTP και OLAP.

## ΕΠΙΛΟΓΟΣ

Στο κεφάλαιο αυτό αναλύθηκε η έννοια Αποθήκη Δεδομένων, γιατί είναι απαραίτητη, πως μπορεί να υλοποιηθεί και σε τι διαφέρει από την παραδοσιακή βάση Δεδομένων. Διερευνήθηκαν οι δυνατότητες εξαγωγής, μετασχηματισμού και φόρτωσης (ETL) με χρήση SSIS. Παρουσιάστηκαν τα βασικά εργαλεία της Microsoft όπως ο SQL Server και το Microsoft SQL Server Management studio. Το Visual Studio με Business Intelligence και η πλατφόρμα SSIS (SQL Server Integration Services). Στο επόμενο κεφάλαιο μελετάμε πως τα δεδομένα που βρίσκονται στην αποθήκη αναλύονται με την τεχνολογία OLAP και τους πολυδιάστατους κύβους στο πλαίσιο του συστήματος υποστήριξης αποφάσεων.

## ΚΕΦΑΛΑΙΟ 2 OLAP

### ΕΙΣΑΓΩΓΗ

Το OLAP περιλαμβάνει πολυδιάστατη ανάλυση των δεδομένων ενός οργανισμού με πολύπλοκους υπολογισμούς για την ανάλυση τάσεων και την παραγωγή γνώσεως υψηλού επιπέδου με σκοπό την λήψη αποφάσεων και την διαμόρφωση στρατηγικής. Δημιουργεί reports για τα διάφορα τμήματα μιας επιχείρησης που ενισχύουν την κατανόηση σημαντικών λειτουργιών της.

Ο όρος OLAP χρησιμοποιήθηκε από τον Edgar F. To 1993 ο οποίος έγραψε τους 12 νόμους της αναλυτικής επεξεργασίας, Codd's twelve rules.

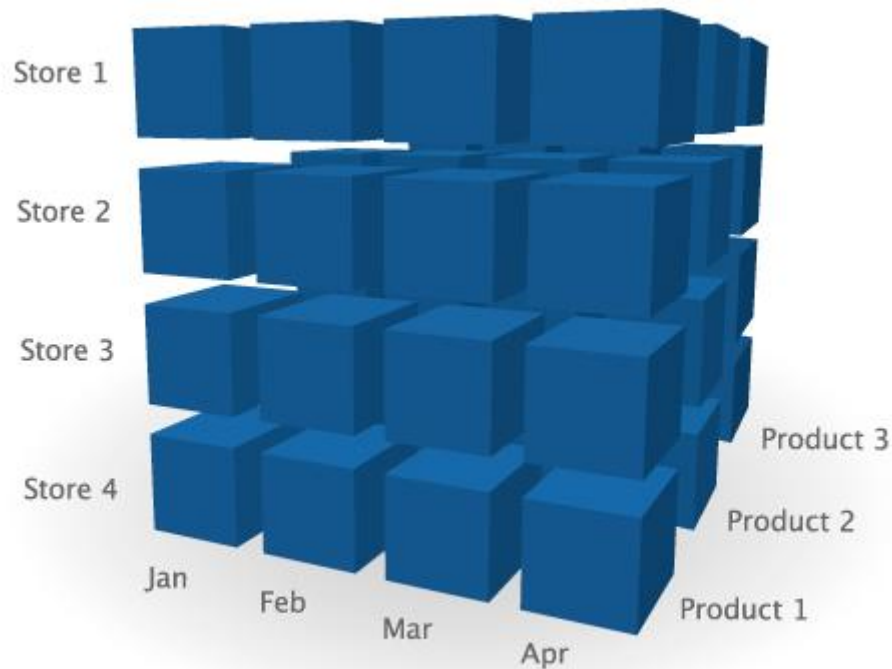
Βασικές λειτουργίες που θα μελετηθούν είναι το roll-up, drill-down και το slicing, dicing.

Πυρήνας ενός OLAP συστήματος είναι ο πολυδιάστατος κύβος (multidimensional cube) ή ο υπέρ-κύβος (hypercube). Θα μελετηθούν οι μετρικές (measures) και οι διαστάσεις (dimensions) του κύβου.

Η Microsoft παρέχει σαν προϊόν το Visual Studio το οποίο με την εγκατάσταση Business intelligence tools είναι ένα εργαλείο υλοποίησης ενός συστήματος υποστήριξης αποφάσεων.

### 2.1 OLAP Cube

Ένας κύβος είναι η δομή δεδομένων για την πολυδιάστατη ανάλυση OLAP, μια πολυδιάστατη βάση δεδομένων. Είναι η μέθοδος για αποθήκευση δεδομένων σε μορφές πολυδιάστατες με σκοπό το γρήγορο Business Intelligence. Αποτελείται από γεγονότα ή μετρικές και από διαστάσεις οι οποίες κατηγοριοποιούν τα γεγονότα. Οι μετρικές προέρχονται συνήθως από τον πίνακα facts στην αποθήκη δεδομένων και οι διαστάσεις από τους πίνακες dimension. Κάθε κελί στο πίνακα dimension αποτελεί μία ιδιότητα. Η γλώσσα προγραμματισμού που χρησιμοποιείται για να γραφούν queries πάνω στο κύβο είναι η MDX (Multidimensional expressions), δημιουργήθηκε από την Microsoft στα τέλη του 1990 και υιοθετήθηκε και από άλλους οργανισμούς που ασχολούνταν με το OLAP.



Εικόνα 12 "OLAP Cube"

Παρότι ο κύβος αποθηκεύει τα δεδομένα παρόμοια με τις παραδοσιακές βάσεις δεδομένων έχει μια πολύ διαφορετική δομή. Οι βάσεις σχεδιάζονται με σκοπό την λειτουργία των εφαρμογών και την αποθήκευση των δεδομένων ενώ ο κύβος σχεδιάζεται με σκοπό την ανάλυση των δεδομένων. Συνεπώς ο σκοπός των κύβων είναι η μεγάλη αποδοτικότητα στα πολύπλοκα ερωτήματα της αναλυτικής επεξεργασίας.

Ένας ακόμη όρος που εμφανίζεται συχνά στο OLAP είναι αυτός του υπέρ-κύβου που προέρχεται από τη γεωμετρία και έχει την σημασία του τετραγώνου με  $n$  διαστάσεις. Ένα χαρακτηριστικό που διαφοροποιεί τους κύβους από τους δισδιάστατους πίνακες είναι η δυνατότητα εποπτείας των δεδομένων από διαφορετικά οπτικά πεδία ή πιο τεχνικά από διαφορετικές διαστάσεις.

Τα στοιχεία των διαστάσεων του κύβου μπορούν να οργανωθούν σαν μια ιεραρχία, ένα σετ από σχέσεις γονέα – παιδιού όπου ο γονέας εμπεριέχει το παιδί. Το παιδί με την σειρά του μπορεί να έχει και αυτό παιδί και αυτό εμβαθύνει στην λεπτομέρεια που μπορεί να επεκταθεί μία διάσταση.

## 2.2 Λειτουργίες OLAP Cube - Operations

Οι βασικές λειτουργίες του κύβου είναι οι εξής :

- Roll up
- Drill Down
- Slice
- Dice
- Pivot

Οι λειτουργίες μας δίνουν την δυνατότητα να πλοηγηθούμε στις ιεραρχίες των διαστάσεων του κύβου να αποκλείσουμε ή να ενσωματώσουμε πληροφορία καθώς και να την οπτικοποιήσουμε κάτω από διαφορετικά οπτικά πεδία.

### 2.2.1 Roll up

Το roll up ή σύμπτυξη δεδομένων περιλαμβάνει την ομαδοποίηση των δεδομένων μέσω της εκτέλεσης ενός βήματος ανόδου στην ιεραρχία της διάστασης. Ο κύβος που προκύπτει μετά την πράξη του roll up εμφανίζει ομαδοποιημένα τα δεδομένα της διάστασης κατά ένα η περισσότερα επίπεδα.

Σε μια διάσταση «τόπος» μπορούμε να ανέβουμε από το επίπεδο πόλη στο επίπεδο νομός και με ακόμη ένα roll-up να ανέβουμε στο επίπεδο χώρα.

### 2.2.2 Drill down

Drill down είναι η αντίστροφη πράξη από το roll up όπου κατεβαίνουμε επίπεδα ιεραρχίας σε μία διάσταση. Ο χρήστης κατεβαίνει επίπεδο από συγκεντρωτικά δεδομένα σε πιο λεπτομερή.

Σε μια διάσταση «τόπος» μπορούμε να κατέβουμε από το επίπεδο χώρα στο επίπεδο νομός και από το επίπεδο νομός στο επίπεδο πόλη.

### 2.2.3 Slice

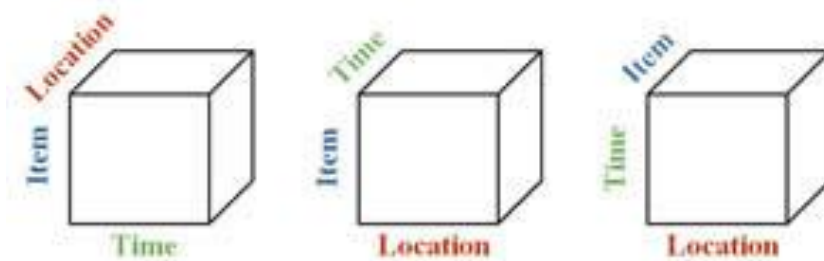
Slice είναι η διαδικασία επιλογής δεδομένων σε μια διάσταση. Ένας κύβος μετά από slice είναι ένα υποσύνολο του κύβου σύμφωνα με μια τιμή ενός επιπέδου διάστασης ή με μια περιοχή τιμών.

### 2.2.4 Dice

Dice είναι η πράξη η οποία δημιουργεί ένα υποσύνολο ή υπό-κύβο επιλέγοντας τις διαστάσεις που θέλουμε να κρατήσουμε. Περιλαμβάνει το σβήσιμο μιας ολόκληρης διάστασης.

### 2.2.5 Pivot

Η πράξη Pivot αναδιατάσσει τον κύβο στους άξονες του δημιουργώντας διαφορετικές προοπτικές για την απεικόνιση του.



Εικόνα 13 "Pivot Operation"

## 2.3 Αρχιτεκτονική SQL Server Analysis Services SSAS



Εικόνα 14 "Microsoft SSAS"

Από πλευράς αρχιτεκτονικής το SSAS είναι ένα ξεχωριστό service που τρέχει σε windows που ακολουθεί την λογική client – server. Κάθε server μπορεί να περιέχει αρκετά analysis services instances. Κάθε instance μπορεί να περιέχει πολλές βάσεις δεδομένων. Οι βάσεις με την σειρά τους μπορεί να περιέχουν



πολυδιάστατες δομές δεδομένων, δομές data mining, data sources και data source views.

Analysis Management Object (AMO) είναι η εφαρμογή λογισμικού που προσδιορίζει, διαχειρίζεται και κάνει publish SSAS κύβους και δομές data mining. Στην πτυχιακή αυτή χρησιμοποιούμε το Business Intelligent Development Studio της Microsoft στο Visual Studio.

Η επικοινωνία γίνεται με το πρωτόκολλο xml για Analysis (XMLA) και όλα τα queries επεξεργάζονται στην μεριά του server.

## 2.4 Microsoft Business Intelligence Development Studio

Στο visual studio είναι το περιβάλλον ανάπτυξης των OLAP cubes. Στην ουσία είναι το Microsoft Visual Studio με εγκατεστημένο το Analysis Services. Το development του κύβου γίνεται στο Visual Studio και αφού γίνει publish ο κύβος μπορεί να επεξεργαστεί στο Visual Studio ή στο Management Studio. Παρακάτω αναφέρουμε κάποια Panel που υπάρχουν στο Visual Studio.

- **Solution Explorer** Αναπαριστά τα διάφορα αντικείμενα που γίνονται developed. Περιέχει data sources, data views, κύβους, διαστάσεις και δομές data mining.
- **Designer** Αφορά την οπτικοποίηση των data sources, των κύβων και των διαστάσεων.
- **Menu** Σχετικά με το project, την βάση, τον κύβο, τις διαστάσεις και το data mining.

## 2.5 Αρχιτεκτονικές αποθήκευσης OLAP

Τα πολυδιάστατα δεδομένα αποθηκεύονται με 3 διαφορετικούς τρόπους Multidimensional OLAP (MOLAP), Relational OLAP (ROLAP) και Hybrid OLAP (HOLAP).

Το MOLAP αποθηκεύει τα δεδομένα σε πολυδιάστατους πίνακες. Πριν δημιουργηθεί ο κύβος τα δεδομένα πρώτα επεξεργάζονται και μετά γεμίζουν τον κύβο. Σε αντίθεση στο ROLAP τα δεδομένα αποθηκεύονται σε σχεσιακές βάσεις δεδομένων συνεπώς δεν χρειάζονται κάποια ιδιαίτερη επεξεργασία. Το ROLAP είναι ένα αφηρημένο επίπεδο πάνω από τα σχεσιακά δεδομένα που παρέχει την δυνατότητα για πολυδιάστατα queries. Τέλος το HOLAP είναι ένα υβρίδιο ανάμεσα στο ROLAP και το MOLAP όπου κατ' επιλογή κάποια δεδομένα μπορεί να είναι σχεσιακά και κάποια πολυδιάστατα.

### 2.5.1 Multidimensional OLAP MOLAP

Το MOLAP υλοποιείται με την επιλογή πολυδιάστατης βάσης δεδομένων και χρησιμοποιούνται πολυδιάστατοι πίνακες. Με αυτό τον τρόπο επιτυγχάνεται γρήγορη εκτέλεση query καθώς η προσπέλαση του κύβου είναι άμεση. Η μεγαλύτερη καθυστέρηση εμφανίζεται στην δημιουργία του κύβου και στο pre processing που χρειάζεται για την δημιουργία του. Πολλές φορές εφαρμόζεται συμπίεση για να μειώσουμε τον αποθηκευτικό χώρο. Το MOLAP περιέχει τα εξής χαρακτηριστικά.

- Υψηλή ταχύτητα
- Λεπτομερείς ιεραρχίες
- Ευρετήρια για ταχεία ανάκτηση
- Ευελιξία στην επέκταση των διαστάσεων
- Μικρός όγκος δεδομένων
- Μεγάλος χρόνος φόρτωσης των δεδομένων

### 2.5.2 Relational OLAP ROLAP

Το σύστημα ROLAP χρησιμοποιείται για ένα σχεσιακό σύστημα διαχείρισης και αποθήκευσης βάσεων δεδομένων όπου υπάρχουν ξεχωριστοί πίνακες γεγονότων και διαστάσεων. Το σύστημα αυτό παρουσιάζει πολύ καλό scalability στη διαχείριση του μεγάλου όγκου των δεδομένων αλλά μειωμένη ταχύτητα στην εκτέλεση των queries. Κάποια χαρακτηριστικά του ROLAP είναι τα εξής.

- Υποστήριξη μεγάλου αριθμού τύπων δεδομένων
- Διαχείριση μεγαλύτερου όγκου δεδομένων
- Παράλληλη επεξεργασία
- Ταυτόχρονη πρόσβαση από πολλούς χρήστες
- Αναγνωρισμένα πρότυπα
- Καθιερωμένη τεχνολογία και κοινώς αποδεκτή
- Εξοικείωση περισσότερων χρηστών με τα χαρακτηριστικά της
- Δεν υποστηρίζει πολυδιάστατους υπολογισμούς
- Τα MDX queries μεταφράζονται σε SQL queries
- Μπορεί να υποστηρίξει real time δεδομένα

### 2.5.3 Hybrid OLAP HOLAP

Το σύστημα HOLAP δημιουργήθηκε με σκοπό να συνδυάσει τις δύο προηγούμενες αρχιτεκτονικές και τα πλεονεκτήματα του καθενός. Κάποιο τμήμα του κύβου μπορεί να αποθηκευτεί με την μορφή MOLAP και κάποιο με την μορφή ROLAP επιτυγχάνοντας αλλού υψηλή ταχύτητα και αλλού υψηλή κλιμάκωση. Το HOLAP ενδεικτικά έχει τα παρακάτω χαρακτηριστικά.

- Ευέλικτη σχεδίαση του τρόπου αποθήκευσης
- Συνδυασμός πλεονεκτημάτων του ROLAP και MOLAP
- Είναι η επικρατούσα τεχνολογία μεταξύ των τριών συστημάτων

### 2.5.4 Σύγκριση ROLAP-HOLAP-MOLAP

Το ROLAP βασίζεται στην υπάρχουσα σχεσιακή τεχνολογία με πλεονεκτήματα για την χρήση έτοιμων προτύπων καθώς και τους περιορισμούς του όπως και την μειωμένη ταχύτητα.

Το MOLAP σε αντίθεση δεν έχει πρότυπα και έχει φτιαχτεί για μεγάλη αποδοτικότητα και ταχύτητα.

Το HOLAP παρέχει ευελιξία στην σχεδίαση συνεπώς μια καλή και προσεκτικά μελετημένη σχεδίαση μπορεί να κρατήσει τα θετικά των υπόλοιπων δύο και να παραμερίσει όσο δυνατόν τα αρνητικά.

Πίνακας 2 "ROLAP – HOLAP - MOLAP"

	<b>ROLAP</b>	<b>HOLAP</b>	<b>MOLAP</b>
<b>Query Performance</b>	Slow	Medium	Fast
<b>Scalability</b>	Best	Medium	Problematic
<b>Data Access</b>	Any Tools	OLAP Only	OLAP Only
<b>Security</b>	Best	Weak	Weak
<b>Storage</b>	Heavy	Moderate	Light

## 2.6 FASMI TEST

Το FASMI είναι ένα σύνολο κανόνων που πρέπει να ικανοποιούνται για να θεωρηθεί ένα OLAP project υψηλού επιπέδου και επιτυχές. Το FASMI είναι ένα ακρωνύμιο για το Fast Analysis of Shared Multidimensional Information.

**Fast Analysis:** Η ανάκτηση των απαντήσεων πρέπει να είναι μέσα σε χρόνο λίγων δευτερολέπτων. Το σύστημα πρέπει να είναι ευέλικτο ώστε αν χρειαστούν καινούριοι υπολογισμοί να είναι εύκολα και γρήγορα υλοποιήσιμοι.

**Shared:** Πρέπει να υποστηρίζεται πολλαπλή προσβασιμότητα, ασφάλεια και ταυτόχρονα ακεραιότητα των δεδομένων.

**Multidimensional:** Το σύστημα πρέπει να παρέχει μια πολυδιάστατη έννοια των δεδομένων με διαφορετικές προοπτικές, με διαστάσεις και ιεραρχίες.

**Information:** Το σύστημα πρέπει να υποστηρίζει, να χειρίζεται και να επεξεργάζεται μεγάλο όγκο δεδομένων.

## 2.7 MDX Multidimensional Expressions

Η MDX είναι μία γλώσσα προγραμματισμού για την δημιουργία query για πολυδιάστατα δεδομένα σε έναν OLAP κύβο. Παρόμοια με την SQL μπορεί να δημιουργήσει queries για σχεσιακές βάσεις δεδομένων. Δημιουργήθηκε το 1997 από την Microsoft από μηχανικούς του SQL Server με προορισμό το Microsoft Analysis Service. Το πρότυπο επικοινωνίας XMLA (XML for Analysis) δημιουργήθηκε το 2001 και εδραίωσε την MDX σαν γλώσσα προγραμματισμού.

Το βασικό και ποιό συχνό MDX query χρησιμοποιεί το «SELECT» το οποίο επιστρέφει ένα αποτέλεσμα το οποίο είναι ένα υποσύνολο από τα πολυδιάστατα δεδομένα που βρίσκονται στο κύβο.

**WITH (Optional):** Το «WITH» επιτρέπει να δημιουργηθεί ένα νέο υπολογισμένο μέλος το οποίο μπορεί να είναι είτε μέτρηση είτε μέλος μιας διάστασης. Το WITH μπαίνει πριν από το SELECT.

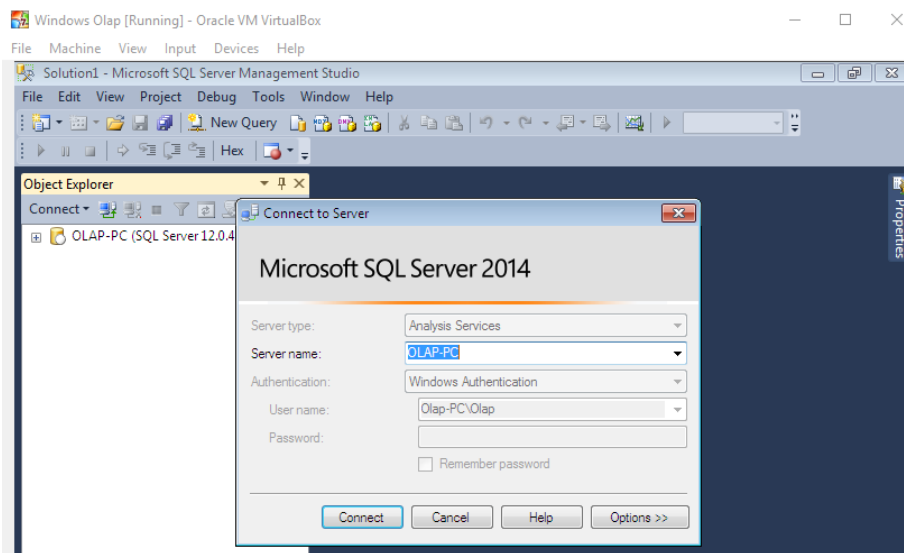
**SELECT:** Το «SELECT» ορίζει τα μέλη των διαφόρων διαστάσεων που θα χρησιμοποιήσουμε στο ερώτημα.

**FROM:** Το «FROM» καθορίζει τον κύβο που θα χρησιμοποιηθεί στο ερώτημα.

**WHERE (Optional):** Το «WHERE» λειτουργεί όπως το “Slice” και περιορίζει τα δεδομένα που θα εμφανιστούν από το ερώτημα κάνοντας slicing ή filtering.

Για να γράψουμε MDX queries θα πρέπει στο Management Studio να συνδεθούμε στην βάση Analysis Services και στον κύβο που δημιουργήσαμε και κάναμε publish στο Visual Studio. Αφού δημιουργήσουμε ένα καινούριο κύβο στο Visual Studio και τον κάνουμε process and run ο κύβος γίνεται deploy στον SQL Server στον Server Analysis Services.

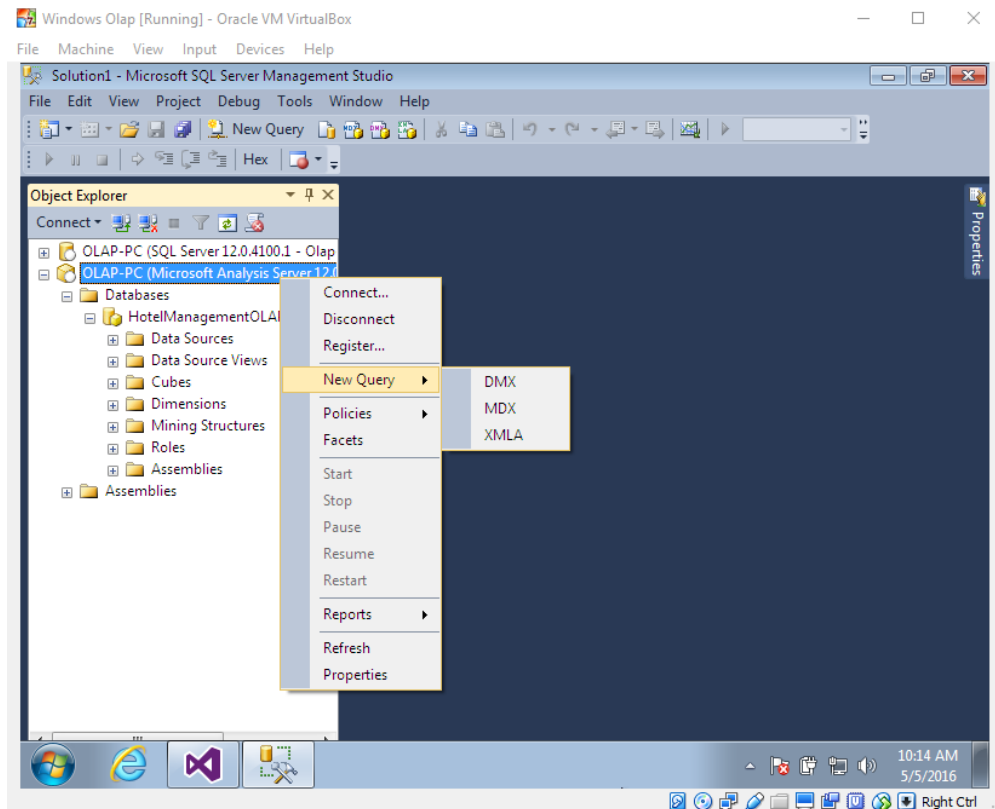
Ο Analysis Services Server είναι ο server που κάνει host τις βάσεις δεδομένων και τους κύβους που ανήκουν στην τεχνολογία OLAP. Στην προκειμένη περίπτωση είναι ο OLAP-PC το οποίο είναι ένα instance του SQL Analysis Server στο εικονικό μηχάνημα μέσα στο Virtual box. Εκεί δημιουργήθηκε η βάση Hotel Management OLAP όταν κάναμε build και deploy το project του κύβου από το Visual Studio. Με αυτό τον τρόπο το project έγινε published και μπορούμε πλέον να έχουμε πρόσβαση στο project μέσα από το SQL Management Studio.



Εικόνα 15 "Connect to Analysis Server Cube"

Ο SQL Server περιέχει την βάση και αυτή με την σειρά της περιέχει τον κύβο με τις μετρικές του και τις διαστάσεις του καθώς και το Data Source και το Data Source View όπως φαίνονται στην επόμενη σελίδα στην εικόνα 16 στον Object Explorer.

Data Source είναι η πηγή δεδομένων από την οποία πήραμε τα δεδομένα που χρειαζόμαστε για να δημιουργήσουμε τον κύβο. Το Data Source είναι το πρώτο πράγμα που δημιουργούμε στο Visual Studio για να παρέχουμε δεδομένα στην εφαρμογή μας. Data Source View είναι η όψη των δεδομένων των πινάκων και των συσχετίσεων μεταξύ τους.



Εικόνα 16 "MDX New Query"

Αφού συνδεθούμε θα εμφανιστεί το project με την βάση και τον κύβο. Επιλέγοντας New Query – MDX θα ανοίξει ο Designer που μπορούμε να γράψουμε MDX queries πάνω στον κύβο που επιθυμούμε.

Όπως βλέπουμε στην εικόνα 16 υπάρχουν τρεις επιλογές για να συνθέσουμε queries, MDX , DMX, XMLA.

Το **MDX** query όπως προαναφέραμε χρησιμοποιεί την γλώσσα MDX για να εξαγει δεδομένα μέσα από ένα SSAS κύβο.

Το **DMX** (Data Mining Extensions) χρησιμοποιείται για SSAS πάνω σε δομές data mining. Είναι μια πιο πολύπλοκη γλώσσα από την MDX. Τα statements του DMX χρησιμοποιούνται για την δημιουργία process, copy, delete, browser και predict σε δεδομένα μοντέλων και δομών data mining. Η Microsoft έχει προμηθεύσει ένα μεγάλο αριθμό από wizard μέσα στα εργαλεία του Business Intelligence για να ελαττώσουν το επίπεδο δυσκολίας της γλώσσας.

Το **XMLA** (XML for Analysis) είναι ένα απλό πρωτόκολλο SOAP (Simple Object Access protocol) και χρησιμοποιείται στο SSAS κύριος για λόγους administration. Για παράδειγμα μπορεί να χρησιμοποιηθεί για backup και restore της βάσης, copy ή move της βάσης ή για να διαβαστούν πληροφορίες meta data.

Δεν θα επεκταθούμε παραπάνω στην γλώσσα MDX καθώς είναι αρκετά κοινή με την SQL και κάποιος που γνωρίζει SQL μπορεί πολύ εύκολα να την κατανοήσει. Εκτός αυτού τα περισσότερα ερωτήματα στο κύβο μπορούν να γίνουν με χρήση του Design και με τεχνικές drag and drop και ο MDX κώδικας παράγεται αυτόματα στο Background. Μόνο σε υπερβολικά σύνθετα ερωτήματα θα χρειαστεί να γράψουμε τα MDX query κατευθείαν με κώδικα.

## ΕΠΙΛΟΓΟΣ

Στο κεφάλαιο αυτό αναλύσαμε τις βασικές δομές της τεχνολογίας OLAP που περιλαμβάνουν σαν βασικό στοιχείο τον κύβο και τις πράξεις που μας βοηθάνε να κατανοήσουμε το περιεχόμενο του κύβου καλύτερα. Μελετήσαμε τους τρόπους αποθήκευσης των δεδομένων μιας πολυδιάστατης δομής, την γλώσσα MDX (Multidimensional Expressions) και τέλος είδαμε τα εργαλεία που παρέχει η Microsoft για το ευρύτερο φάσμα της τεχνολογίας OLAP.

## ΚΕΦΑΛΑΙΟ 3 Τεχνικά παραδείγματα

### ΕΙΣΑΓΩΓΗ

Σε αυτό το κεφάλαιο παρουσιάζουμε την χρήση των διάφορων εργαλείων της Microsoft για να δημιουργήσουμε και να επεξεργαστούμε την βάση δεδομένων, την αποθήκη δεδομένων, το SSIS που θα χειριστεί το ETL από την βάση στην αποθήκη και τέλος την δημιουργία του κύβου καθώς και την χρήση των πράξεων OLAP.

Για την βάση δεδομένων και την αποθήκη δεδομένων χρησιμοποιούμε το SQL Management Studio που βρίσκεται στον SQL Server.

Για το SSIS και το SSAS χρησιμοποιούμε το Visual Studio με Business Intelligence Tools.

Τέλος τον κύβο μπορούμε να τον αναλύσουμε και να τον μελετήσουμε τόσο στο SQL Management Studio και στο Visual Studio καθώς και στο Microsoft Office Excel.

#### 3.1 Η βάση δεδομένων Hotel Management

Η βάση δεδομένων Hotel Management δημιουργήθηκε στο SQL Management studio 2014 και γίνεται host από τον SQL Server. Όπως αναφέραμε στο κεφάλαιο 1 η βάση διαχειρίζεται ένα σύστημα κρατήσεων πολλαπλών ξενοδοχείων ενός οργανισμού τουρισμού. Οι πίνακες και οι συσχετίσεις τους φαίνονται αναλυτικά στην εικόνα 3 του 1<sup>ου</sup> κεφαλαίου.

#### 3.2 Η Αποθήκη Δεδομένων Hotel Management DW

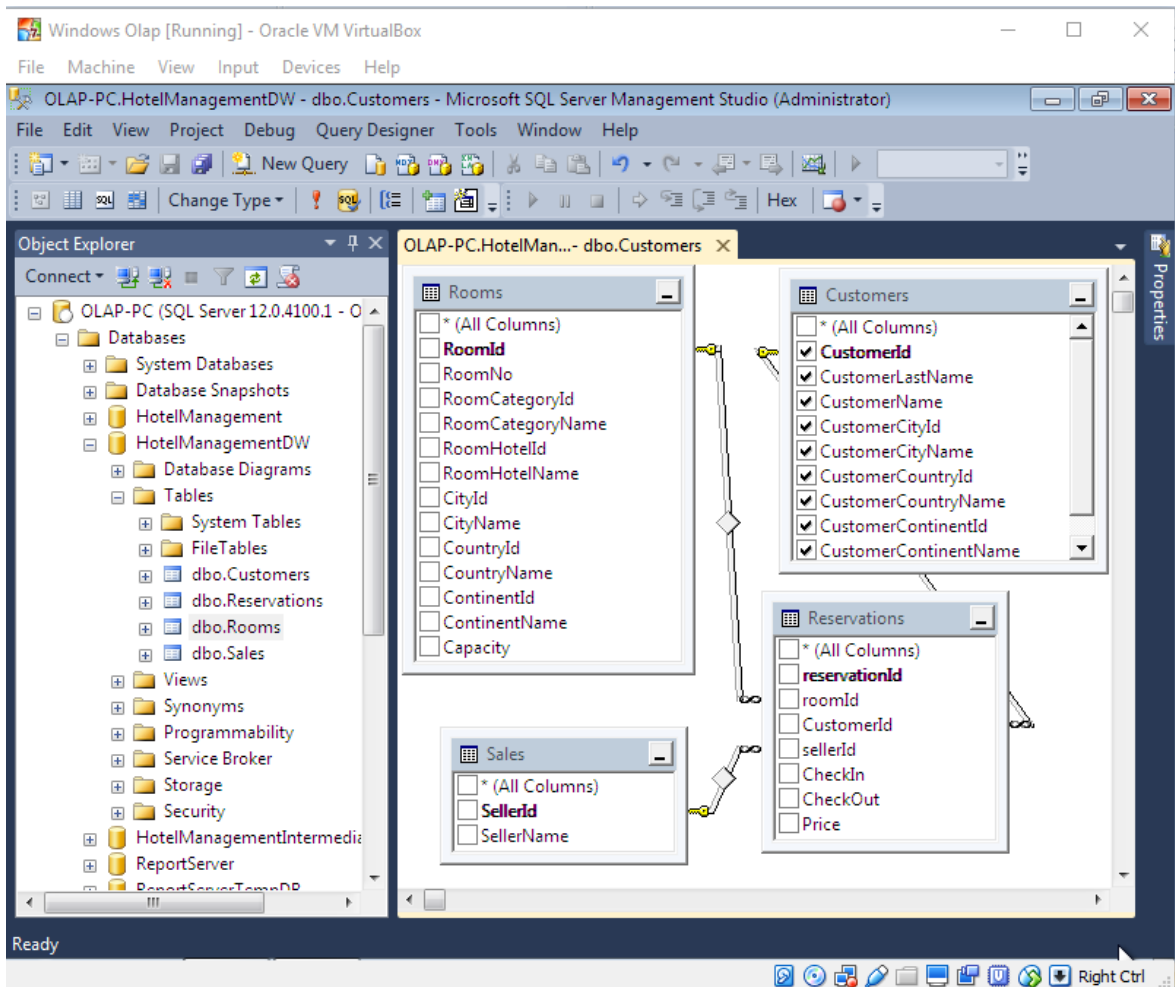
Η αποθήκη δεδομένων Hotel Management DW δημιουργήθηκε στον SQL Management Studio 2014 και κρατάει τα δεδομένα της βάσης Hotel Management που είναι αποκλειστικά διαμορφωμένα και για χρήση από το σύστημα OLAP.

Η αρχιτεκτονική που χρησιμοποιήθηκε είναι το star schema με ένα πίνακα γεγονότων «reservations» και γύρω του τις διαστάσεις «sales», «customer», «rooms». Οι πίνακες διαστάσεων απλοποιήθηκαν όσον αφορά τις συσχετίσεις τους και η πληροφορία ενσωματώθηκε μέσα στους ίδιους πίνακες.



Όπως βλέπουμε στην εικόνα 17 ο πίνακας rooms ενσωμάτωσε την πληροφορία του πίνακα hotels, cities, countries και continents ενώ στην βάση συνδέονταν με τους εκάστοτε πίνακες με foreign keys. Με αυτό τον τρόπο επιτυγχάνουμε την απλοποίηση των ιεραρχιών σε ένα πίνακα και δεν χρειάζονται joins με άλλους πίνακες για να πάρουμε πληροφορία.

Για παράδειγμα στην βάση δεδομένων Hotel Management αν θέλαμε να βρούμε σε πια χώρα ανήκει ένα δωμάτιο θα έπρεπε να κάνουμε inner join τους πίνακες rooms με hotels, hotels με cities και τέλος inner join cities με countries για να φέρουμε σε ένα select το countryName ενός δωματίου. Αντιλαμβανόμαστε τι φόρτο παράγει αυτό το query όταν ψάχνει σε χιλιάδες ή εκατομμύρια εγγραφές.



Εικόνα 17 "Αποθήκη Δεδομένων Hotel Management DW"

Τώρα πλέον ο πίνακας rooms περιέχει όλη την πληροφορία των υπόλοιπων πινάκων και είναι μια ολοκληρωμένη διάσταση. Το ίδιο και ο πίνακας customers. Ο πίνακας sales δεν χρειάστηκε αλλαγή ούτε και ο πίνακας reservations που είναι ο πίνακας facts. Το σχήμα αστέρα δημιουργήθηκε με τις συσχετίσεις των διαστάσεων με τον πίνακα γεγονότων και η αποθήκη είναι έτοιμη να δεχθεί τα δεδομένα.

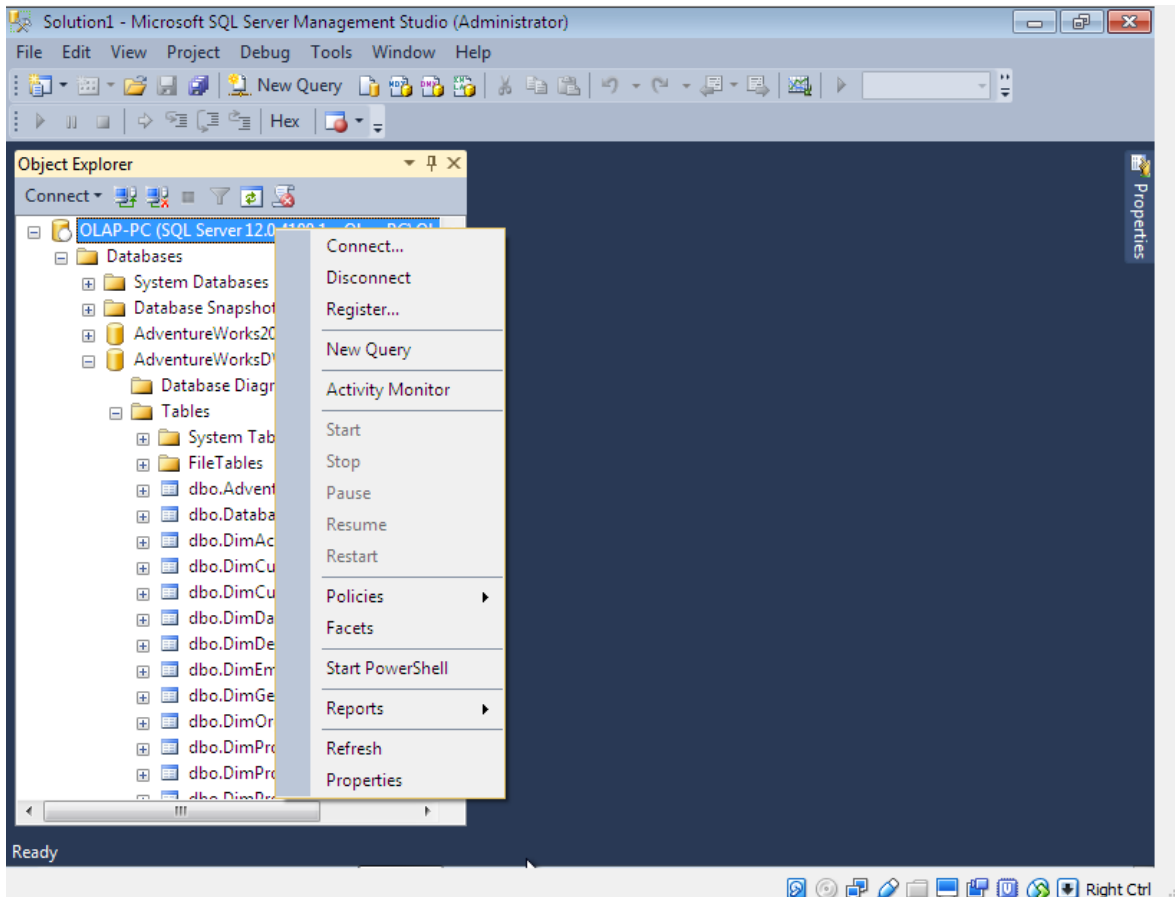
### 3.3 Η βάση δεδομένων Adventure Works

Η Adventure Works Cycle είναι μια φανταστική πολυεθνική κατασκευαστική εταιρία που εμπορεύεται ποδήλατα σε διάφορες χώρες. Για την εταιρία αυτή έχει φτιαχτεί η βάση δεδομένων Adventure Works. Η βάση δεδομένων υποστηρίζει τις λειτουργίες των πληροφοριακών συστημάτων της εταιρίας. Θα μπορούσαμε να υποθέσουμε ότι αυτή είναι η βάση ενός ERP (Enterprise Resource Planning) συστήματος της εταιρίας. Η βάση Adventure Works χωρίζεται σε πέντε κύριες κατηγορίες Human Resources, Person, Production, Purchasing και Sales.

- **Human Resources** Αυτή η κατηγορία περιλαμβάνει τους υπαλλήλους της εταιρίας.
- **Person** Σε αυτή την κατηγορία βρίσκονται τα ονόματα και οι διευθύνσεις από τους πελάτες, τους προμηθευτές και τους υπαλλήλους.
- **Production** Αυτή η κατηγορία περιλαμβάνει τα προϊόντα που παρασκευάζονται και πουλιούνται από την εταιρία.
- **Purchasing** Στο purchasing υπάρχουν δεδομένα για τους προμηθευτές και τα προϊόντα που αγοράζονται από την εταιρία.
- **Sales** Στο Sales υπάρχουν δεδομένα σχετικά με τις πωλήσεις και τους πελάτες.

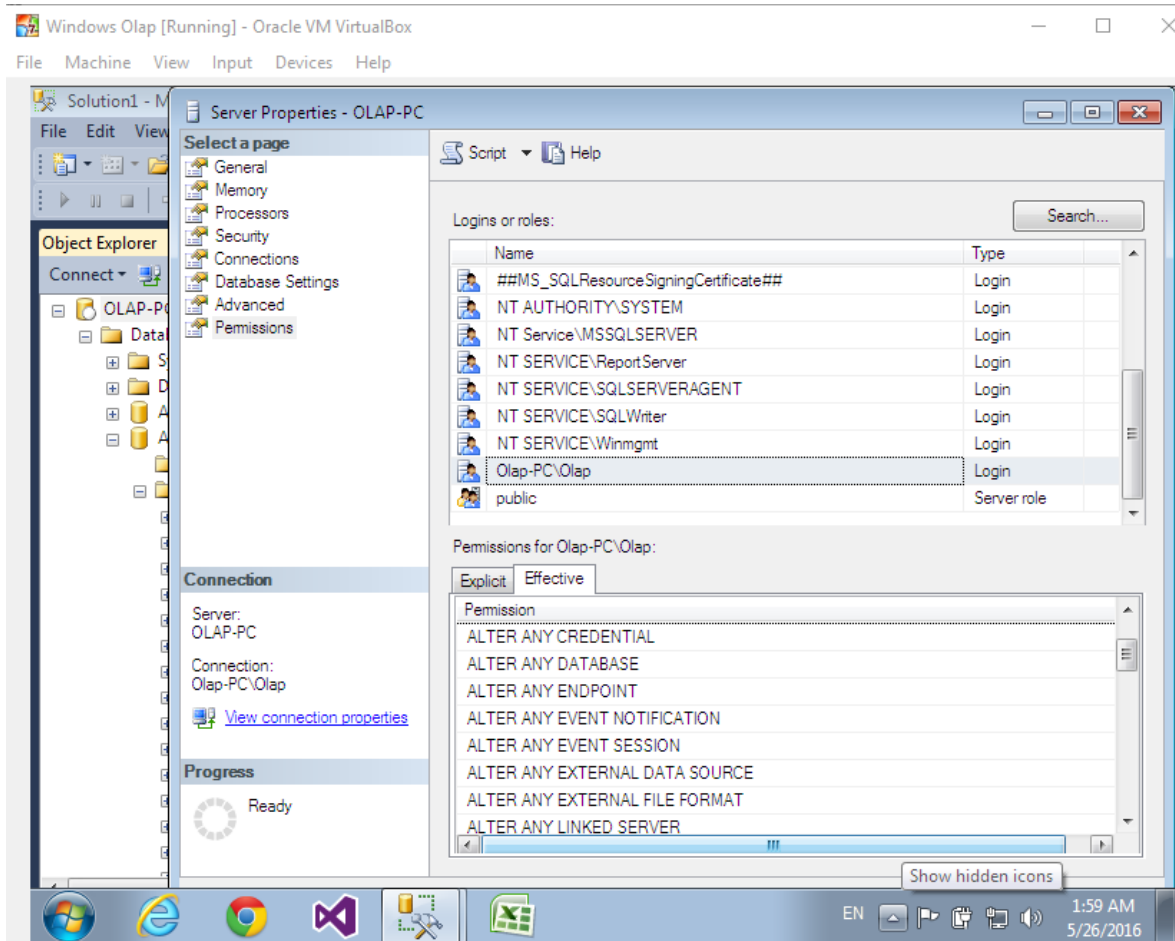
Το backup αρχείο της βάσης «AdventureWorks.bak» το κατεβάσαμε από την ιστοσελίδα Codeplex και το κάναμε restore στον SQL Server Management Studio για να μπορούμε να χρησιμοποιήσουμε την βάση.

Μια σημαντική παρατήρηση που πρέπει να κάνουμε σε αυτό το σημείο, όπως βλέπουμε στην εικόνα 18 και 19, είναι να βεβαιωθούμε ότι έχουμε όλα τα permissions για τον χρήστη μας για την βάση στον SQL Management Studio και στα folders που χρειάζονται access. Το ίδιο ισχύει σε όλες τις βάσεις δεδομένων και τις αποθήκες δεδομένων.



Εικόνα 18 "βάση δεδομένων properties"

Στον server των βάσεων δεδομένων όπως βλέπουμε στην εικόνα 18 πηγαίνουμε στα properties. Ύστερα πηγαίνουμε στο μενού permissions και στον χρήστη OLAP-PC/Olap όπως φαίνεται στην εικόνα 19. Στο tab effective βλέπουμε τι δικαιώματα είναι σε ισχύ πάνω στις βάσεις του συγκεκριμένου server.



Εικόνα 19 "βάση Δεδομένων properties permissions"

Σε περίπτωση που θέλουμε να προσθέσουμε κάποιο δικαίωμα πηγαίνουμε στο explicit και κάνουμε check στο column Grant στο δικαίωμα που θέλουμε να παραχωρήσουμε στον χρήστη.

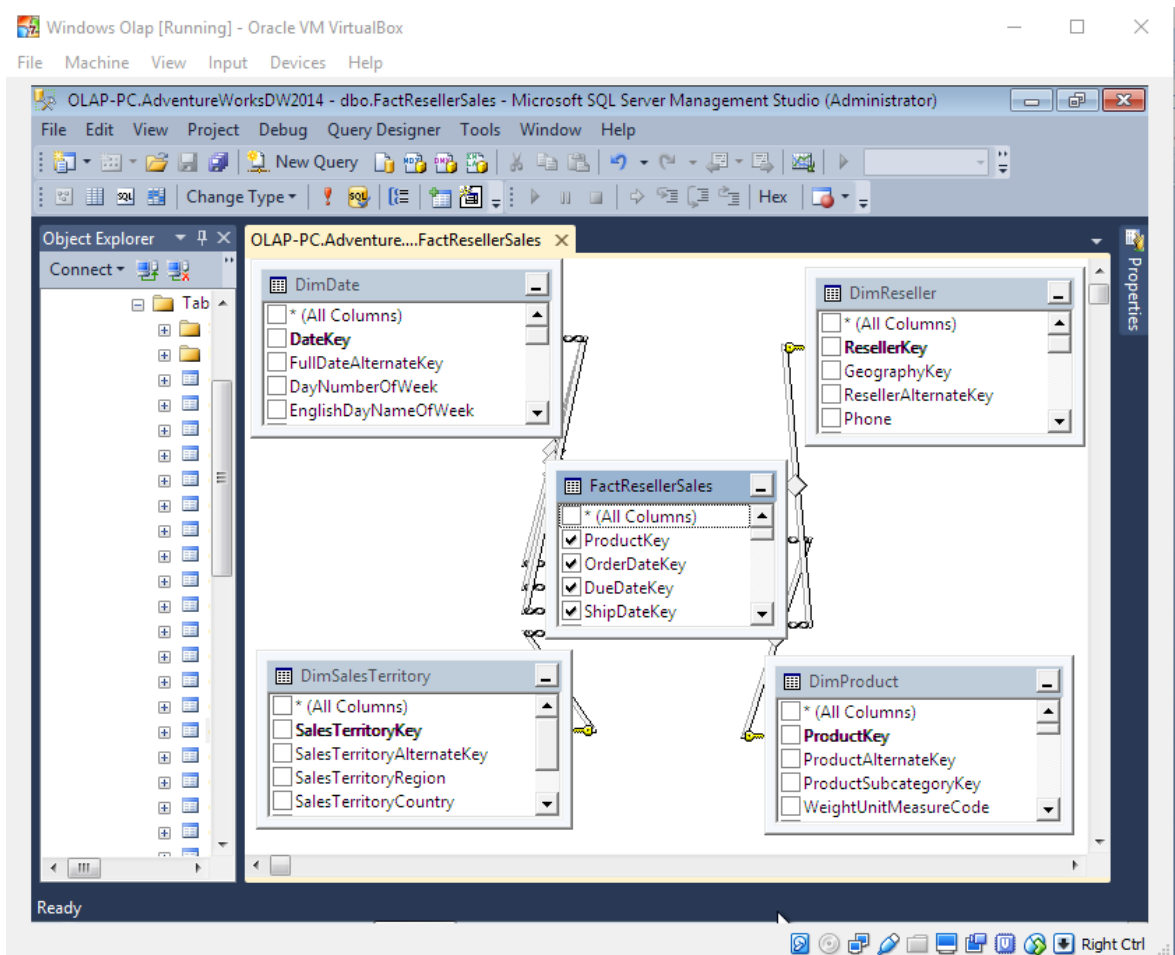
### 3.4 Η Αποθήκη δεδομένων Adventure Works DW

Η αποθήκη δεδομένων Adventure Works DW δημιουργήθηκε από την βάση Adventure Works με σκοπό να εφαρμόσουμε Business Intelligence για την πάνω στα δεδομένα της εταιρίας. Τα δεδομένα στην αποθήκη υποστηρίζουν τόσο data mining όσο και OLAP. Στην περίπτωση μας θα εξετάσουμε την τεχνολογία OLAP. Η αποθήκη αυτή είναι ένα υποσύνολο της βάσης Adventure Works και τα δεδομένα είναι συγχρονισμένα με την βάση.

Το backup αρχείο της βάσης «AdventureWorksDW.bak» το κατεβάσαμε από την ιστοσελίδα Codeplex και το κάναμε restore στον SQL Server Management Studio για να μπορούμε να χρησιμοποιήσουμε την βάση.

Στην αποθήκη δεδομένων έχουμε πίνακες fact και πίνακες dimension που αντιστοιχούν στην βάση δεδομένων.

Στην εικόνα 20 μπορούμε να δούμε ένα σχήμα αστέρα της αποθήκης δεδομένων Adventure Works DW που δημιουργήσαμε στο diagram pane του Management Studio. Βλέπουμε τον πίνακα γεγονότων FactResellerSales που περιέχει πληροφορία για τις πωλήσεις που έγιναν από τους μεταπωλητές της εταιρίας. Οι διαστάσεις DimDate, DimReseller, DimSalesTerritory και DimProduct συνδέονται με το πίνακα γεγονότων με foreign key και προσδίδουν πληροφορία στην πώληση για την ημερομηνία που έγινε, το ποιός έκανε την πώληση, την περιοχή που έγινε η πώληση και τέλος το προϊόν που πωλήθηκε. Σε επόμενο κεφάλαιο θα δούμε πως μπορεί να δημιουργηθεί ο κύβος πάνω σε αυτό το schema ώστε να εξάγουμε σημαντική πληροφορία από την πώληση των προϊόντων.

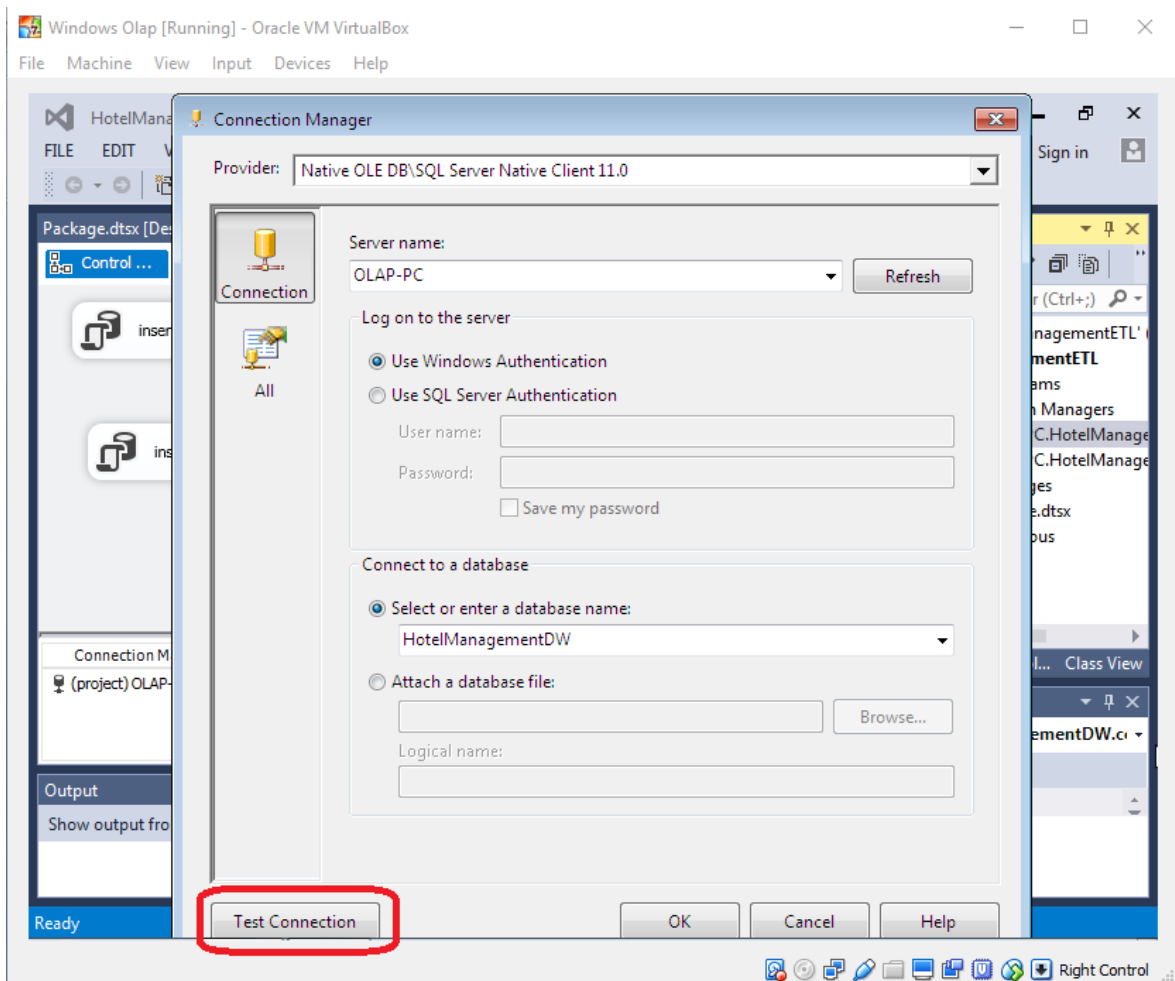


Εικόνα 20 "Star Schema Adventure Works DW"

### 3.5 SSIS πακέτο ETL Hotel Management και Customer Integration

Στο Microsoft visual studio 2013 δημιουργήσαμε ένα καινούριο project Integration Services Hotel Management ETL. Σε αυτό το project θα γράψουμε τα SQL scripts

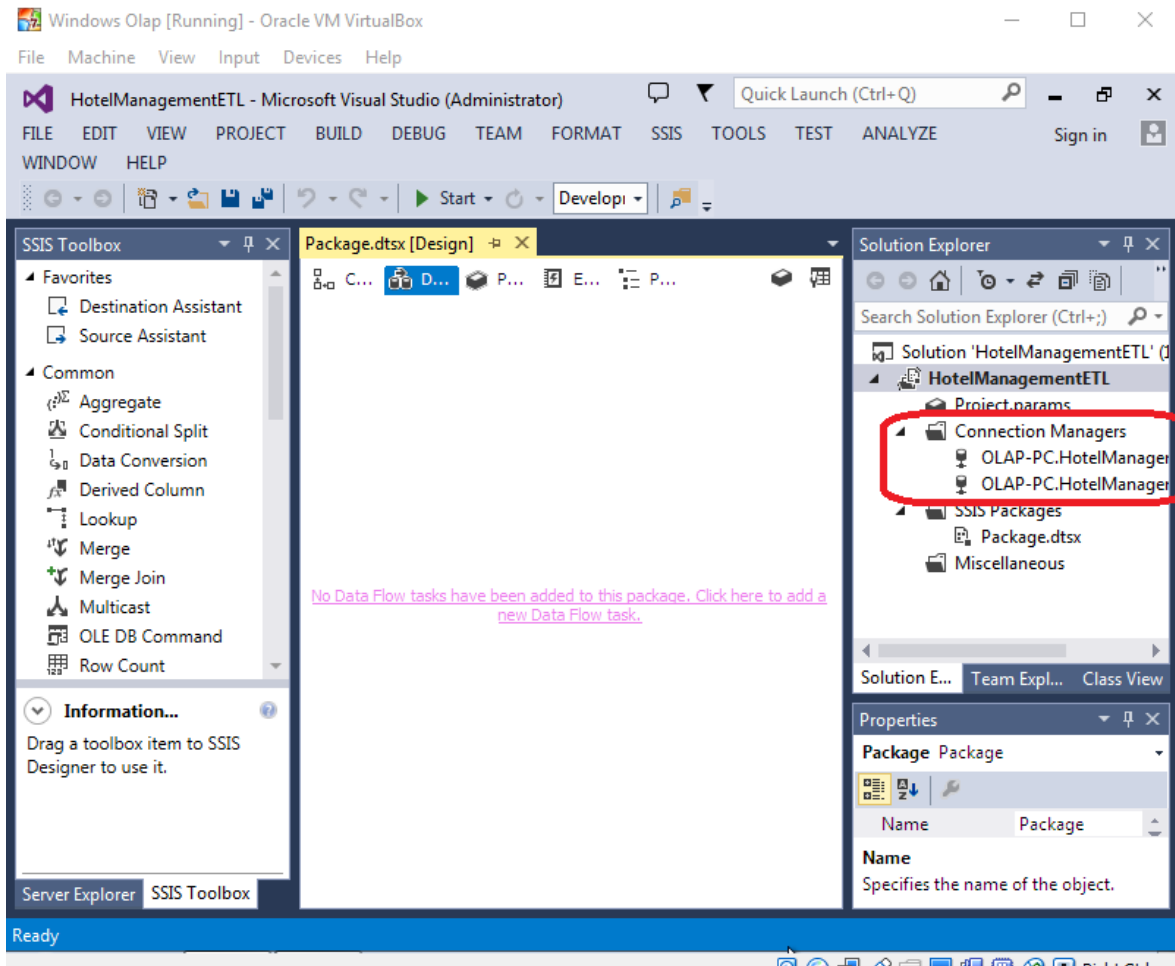
που θα μεταφέρουν τα δεδομένα από την βάση δεδομένων στην αποθήκη. Στην περίπτωση μας επειδή βρισκόμαστε σε test περιβάλλον και η βάση δεδομένων δεν χρησιμοποιείται από κάποια εφαρμογή, θα φορτώσουμε τα δεδομένα από την βάση κατευθείαν στην Αποθήκη. Συνιστάται όμως σε live και highly transactional εφαρμογές με μεγάλο φόρτο να χρησιμοποιούνται «μεσαίες – intermediate βάσεις» που θα στεγάζουν τα δεδομένα και θα βοηθήνε στην γρήγορη απελευθέρωση την κύριας βάσης.



Εικόνα 21 "SSIS Connection Manager"

Αφού δημιουργήσουμε το project θα προσθέσουμε δύο connection managers, ένα για την βάση Hotel Management και ένα για την βάση Hotel Management DW. Το Connection Manager είναι το πρώτο πράγμα που κάνουμε σε κάθε καινούριο Integration Services Project. Αφορά την σύνδεση του project με τα δεδομένα που θα χρησιμοποιήσουμε για αυτό. Καθορίζει σε ποιόν server θα συνδεθούμε, τα credentials που χρειάζονται καθώς και την βάση που συνδεθούμε και θα χρησιμοποιήσουμε. Στην περίπτωση μας στο πρώτο Connection Manager συνδεόμαστε στον SQL Server και την βάση από τον οποίο θα τραβήξουμε τα δεδομένα. Στο δεύτερο connection συνδεόμαστε στον SQL Server και στην βάση

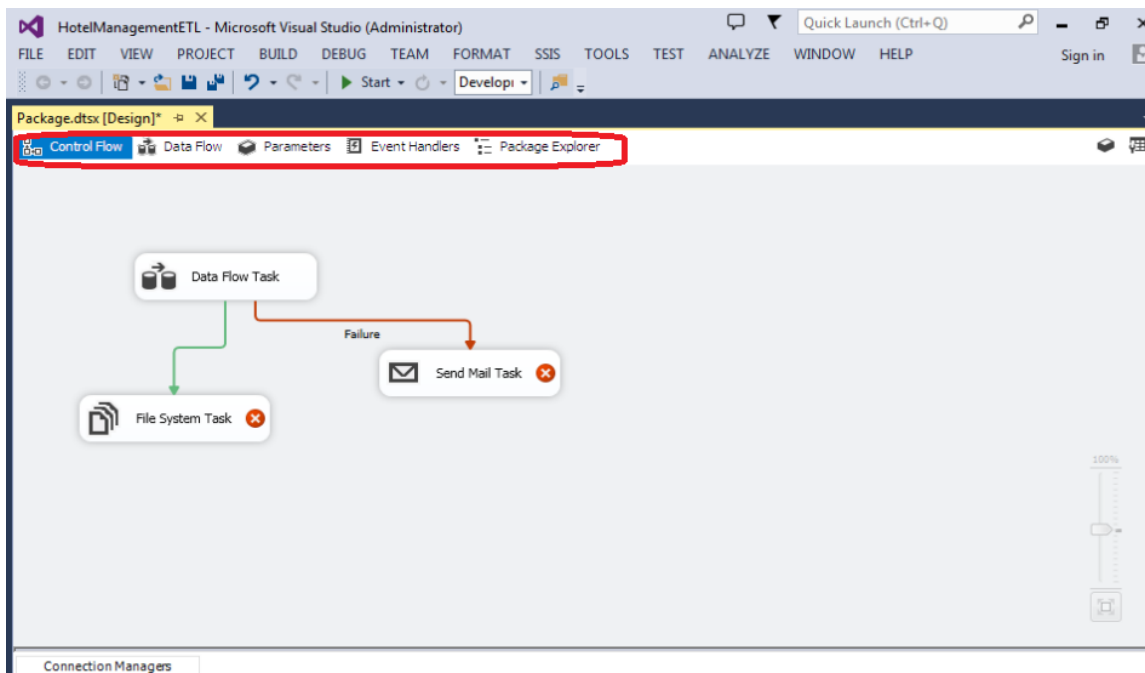
που θα βάλουμε τα δεδομένα. Οι βάσεις που θα χρησιμοποιήσουμε βρίσκονται στον ίδιο SQL Server που έχουμε τοπικά στο μηχάνημα στο Virtual Box συνεπώς χρησιμοποιούμε τον ίδιο Server με τα ίδια credentials και αλλάζουμε μόνο την βάση δεδομένων. Μετά την δημιουργία του Connection Manager πατάμε το κουμπί Test Connection για να δούμε αν μπορεί επιτυχώς να συνδεθεί στον Server και την βάση που ορίσαμε όπως φαίνεται στην εικόνα 21.



Εικόνα 22 "SSIS Connection Managers"

Πριν προχωρήσουμε στο project Hotel Management ETL και στην εισαγωγή των execute SQL Tasks στο project θα πρέπει να αφιερώσουμε λίγο χρόνο για να εξηγήσουμε κάποιες λειτουργίες του SSIS στο Visual Studio που αφορούν το Panel Design ενός πακέτου dtsx.

Στο Design του πακέτου έχουμε πέντε κύρια μενού το Control Flow, το Data Flow, το Parameters, το Event Handlers και το Package Explorer αυτά θα τα αναλύσουμε περισσότερο στην εφαρμογή Customers Integration Hotel Management εμβαθύνοντας κυρίως στο Control Flow και στο Data Flow.



Εικόνα 23 "SSIS Design Menu"

Το **Control Flow** καθορίζει την ροή εκτέλεσης των διάφορων tasks που υπάρχουν μέσα του. Το κάθε task συνδέεται με το επόμενο ενώνοντας το με την χρήση μιας πράσινης γραμμής. Ένα task μπορεί να εκτελεστεί επιτυχώς ή ανεπιτυχώς. Αν το task εκτελέστηκε επιτυχώς ακολουθεί την σύνδεση με την πράσινη γραμμή και πηγαίνει στο επόμενο task που το έχουμε συνδέσει. Αν το task εκτελέστηκε ανεπιτυχώς, πράγμα το οποίο μπορεί να συμβεί για διάφορους λόγους, και έχει συνδεθεί με κόκκινη γραμμή με κάποιο άλλο task ακολουθεί το connection με την κόκκινη γραμμή και πηγαίνει να εκτελέσει το επόμενο task.

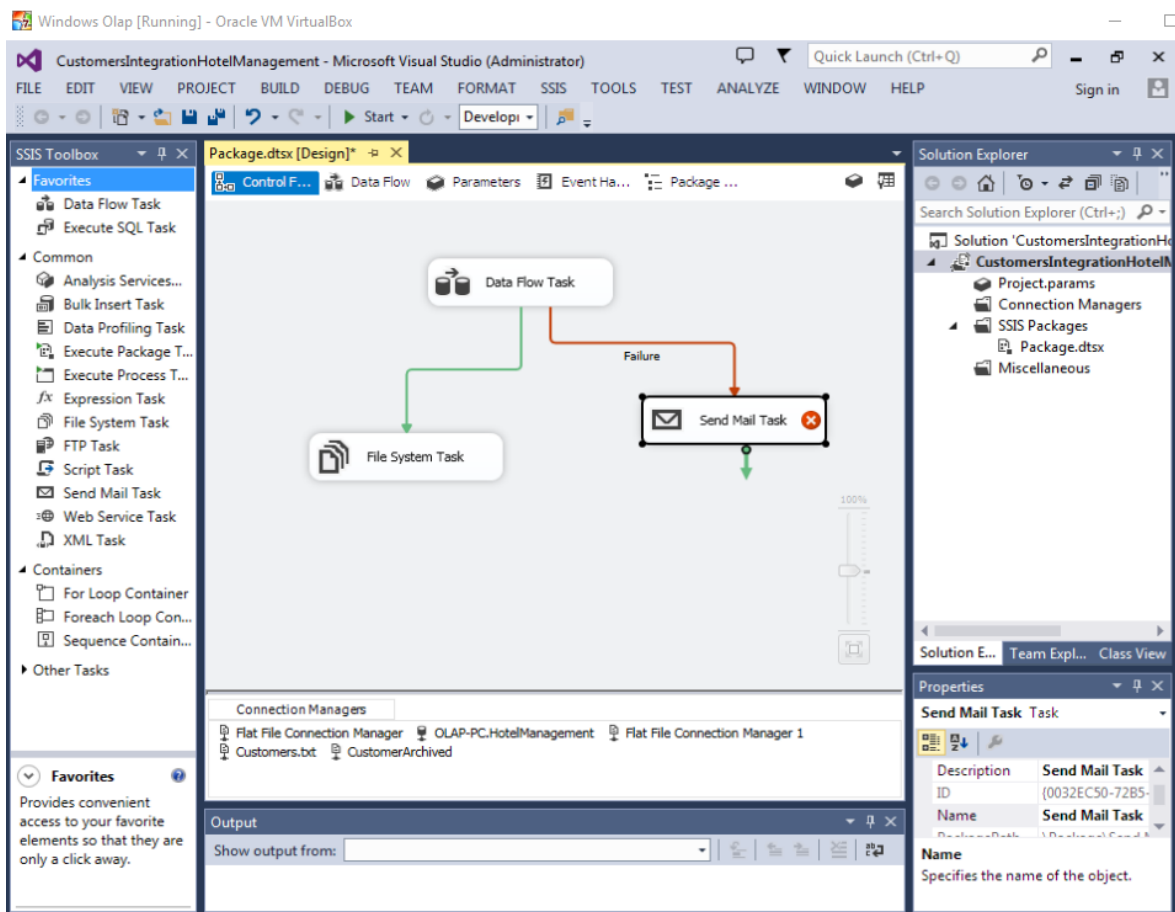
Στην εικόνα 24 βλέπουμε ένα παράδειγμα για να επεξηγήσουμε κάποιες λειτουργίες και δυνατότητες του Control flow. Ας υποθέσουμε ότι ένα ταξιδιωτικό πρακτορείο που συνεργάζεται με τα ξενοδοχεία της εφαρμογής Hotel Management μας αποστέλλει κάθε φορά που έχει καινούριους πελάτες μια λίστα με τους καινούριους πελάτες σε ένα coma delimited file το οποίο πρέπει να περάσουμε στην βάση στον πίνακα customers.

Δημιουργήσαμε ένα project SSIS για να εισάγουμε αυτά τα δεδομένα στην βάση Hotel Management. Στο control flow της εικόνας 24 του project χρησιμοποιούμε στην αρχή ένα data flow task για να διαβάσει τα δεδομένα από το coma delimited file Customers.txt να τα επεξεργαστεί και να τα αποστείλει στην βάση. Αν το data flow task εκτελεστεί επιτυχώς τότε στην συνέχεια το file system task που είναι συνδεδεμένο θα εκτελεστεί. Το file system task μεταφέρει το αρχείο που διαβάστηκε σε ένα φάκελο archive «Customer Archived» και τελειώνει η εφαρμογή.



Σε περίπτωση που στο data flow υπήρξε failure θα πάει να εκτελέσει το send mail task. Αυτό με την σειρά του χρησιμοποιεί ένα mail server για να μας στείλει email ότι το πακέτο SSIS αντιμετώπισε ένα σφάλμα και δεν εκτελέστηκε.

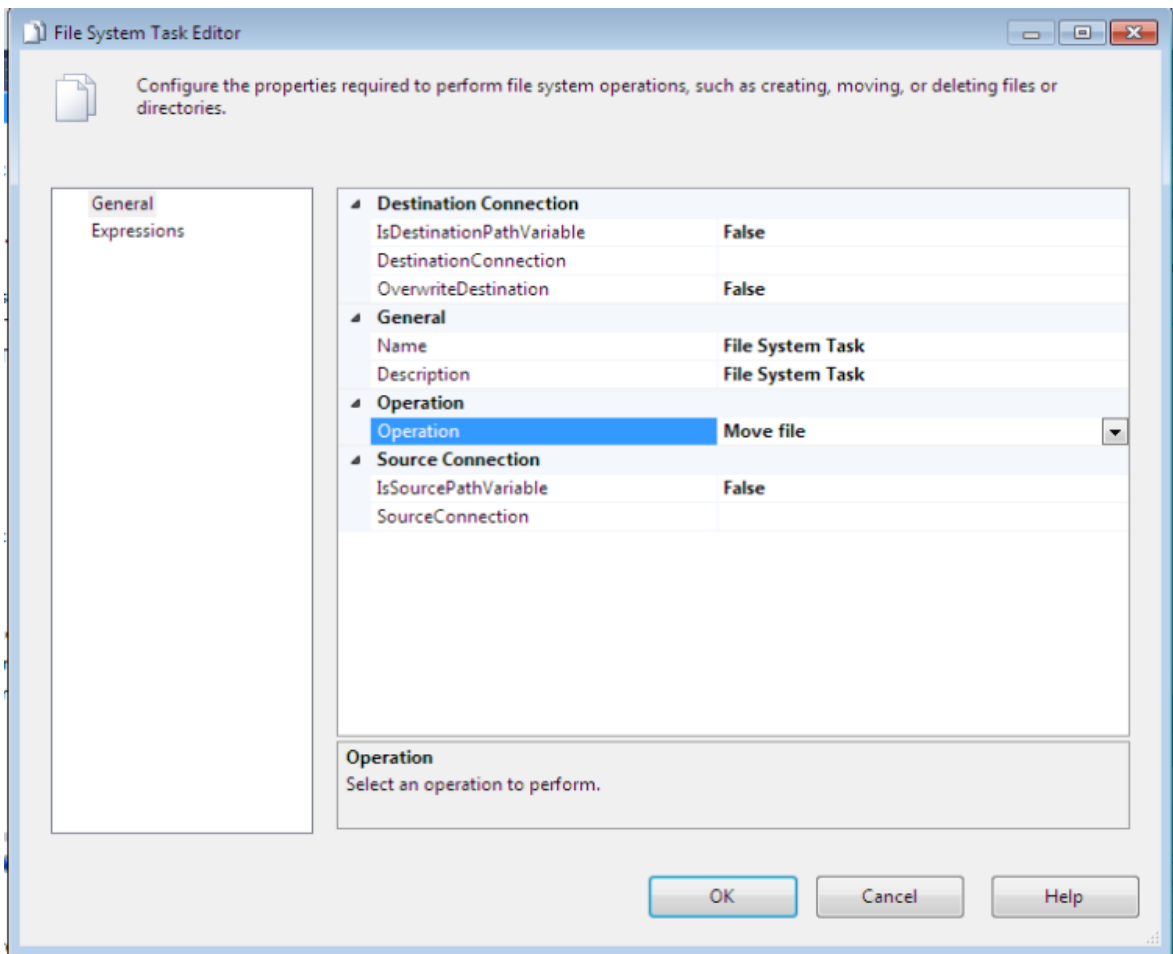
Το send mail Task έχει την δυνατότητα αφού οριστεί ένας smtp Mail Server να αποστείλει το email που θα του ορίσουμε στην διεύθυνση που θα του ορίσουμε όταν εκτελεστεί το send mail task. Στο παράδειγμα μας το χρησιμοποιούμε απλά για να το παρουσιάσουμε σαν εργαλείο του toolbox και όχι για να αποστείλουμε κάποιο email καθώς δεν έχουμε κάποιον Mail Server.



Εικόνα 24 "Παράδειγμα Customers Integration Hotel Management"

Να επισημάνουμε ότι αν το task δεν έχει συνδεθεί με κόκκινη γραμμή και συμβεί κάποιο σφάλμα που προκαλέσει την ανεπιτυχή εκτέλεση του παγώνει και σταματάει το project. Μόλις τελειώσει το πρώτο task επιτυχώς ξεκινάει το επόμενο που είναι άμεσα συνδεδεμένο με το task που τελείωσε, το ίδιο ισχύει και για τα επόμενα task που υπάρχουν μέσα στο work flow.

Το control flow δεν επεξεργάζεται τα δεδομένα άμεσα από μόνο του, αλλά οργανώνει και εκτελεί τα διάφορα tasks που περιέχει.



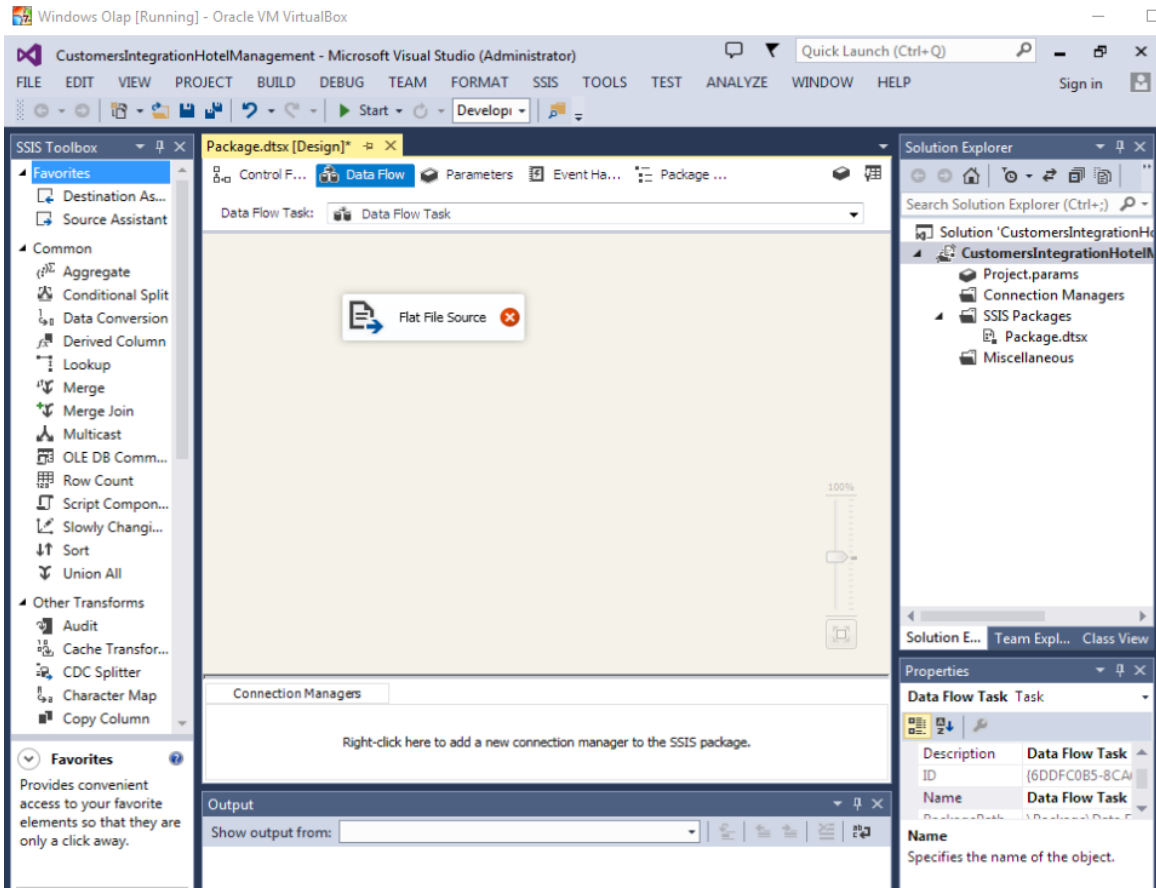
Εικόνα 25 "File System Task"

Το File System Task είναι ένα εργαλείο του SSIS που αναλαμβάνει λειτουργίες όπως copy, move, delete, rename είτε σε κάποιο file είτε σε κάποιο directory. Αυτό ορίζεται στο tab operation στο δικό μας παράδειγμα θα επιλέξουμε το operation move file. Στο «destination connection» ορίζουμε από που θέλουμε στο δίσκο να μεταφέρει το αρχείο, στην περίπτωση μας θα ορίσουμε στο φάκελο Customers Archived. Στο source connection ορίζουμε από που θα βρει το αρχείο στο δίσκο, στο παράδειγμα μας θα πάρει το αρχείο από το desktop. Στην εικόνα 25 βλέπουμε τον Editor του File System Task.

Στο control flow μπορούμε να κάνουμε drag and drop από το SSIS toolbox ένα data flow task. Κάνοντας διπλό click πάνω στο Data Flow Task πηγαίνουμε στο επόμενο μενού του Design, μέσα στο Data Flow που θα εξηγήσουμε παρακάτω.

Το **Data Flow** είναι το μενού του design στο οποίο τα δεδομένα εξάγονται, επεξεργάζονται ή μετασχηματίζονται και τέλος φορτώνονται. Τα δεδομένα εξάγονται από διάφορα data sources όπως OLE DB Source, Raw File Source, Flat File Source, Excel.

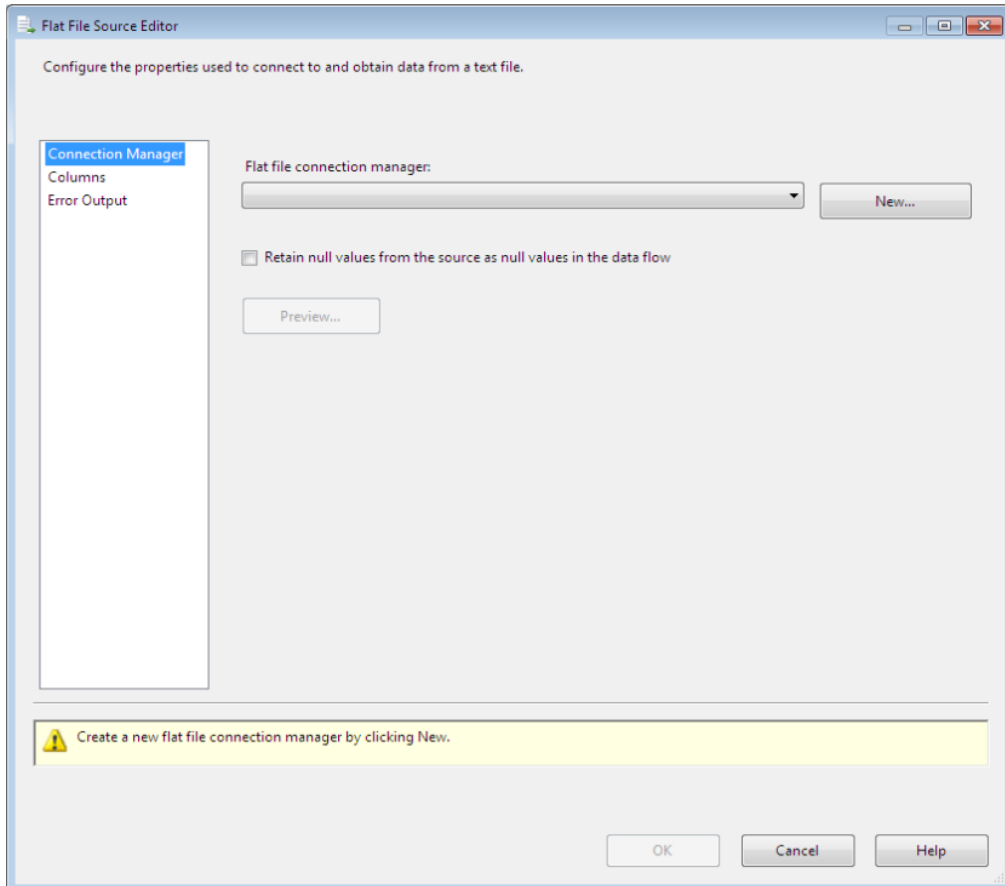
Αφού έχουμε φορτώσει τα δεδομένα από το source χρησιμοποιούμε διάφορα data flow transformations όπως Data Conversion, Derived Column, Merge για να επεξεργαστούμε τα δεδομένα ανάλογα με τις ανάγκες του project. Τέλος χρησιμοποιούμε διάφορα data flow destinations αντίστοιχα με αυτά στα data sources για να φορτώσουμε τα δεδομένα εκεί που επιθυμούμε.



Εικόνα 26 "Flat File Source"

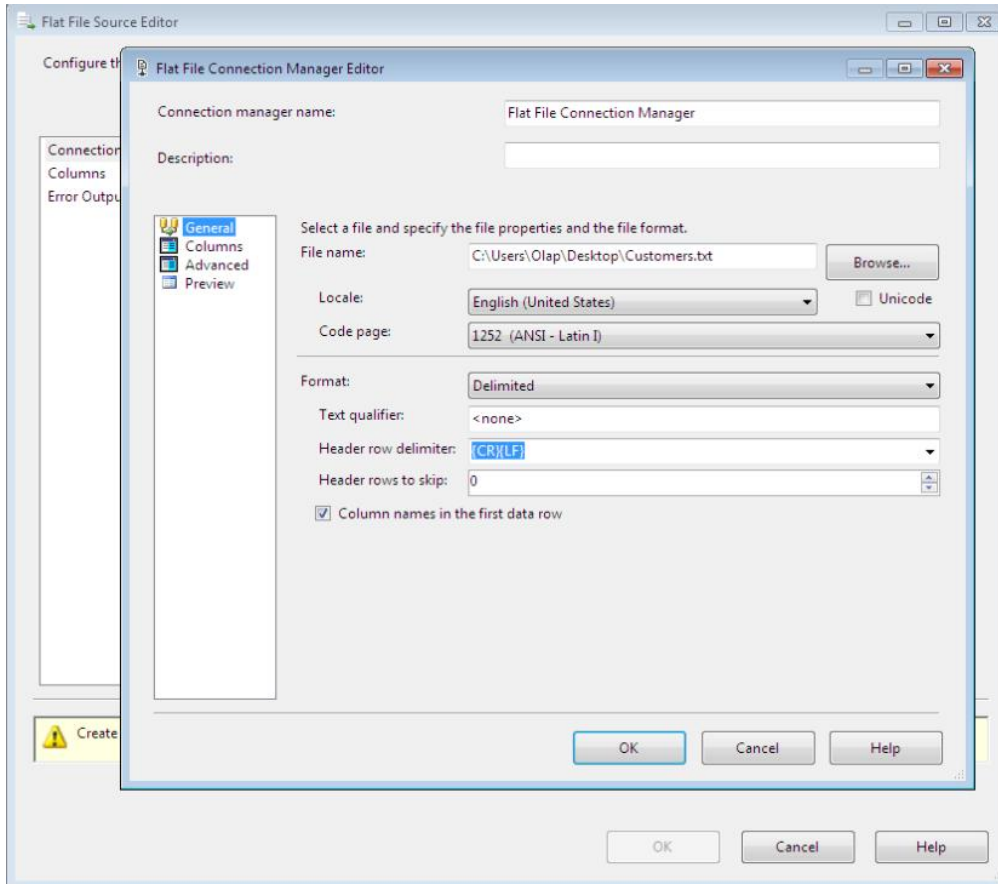
Στο παράδειγμα με τους πελάτες του πρακτορείου κάνοντας διπλό click στο Data flow task μέσα στο control flow μας μεταφερόμαστε στο μενού Data Flow, εκεί έχουμε κάνει drag and drop ένα flat file source που θα αναλάβει να διαβάσει το Customers.txt.

Κάνοντας διπλό click στο flat file source που βλέπουμε στην εικόνα 27 μας ανοίγει τον wizard για να ρυθμίσουμε το flat file source.



Εικόνα 27 "Flat File Source Editor"

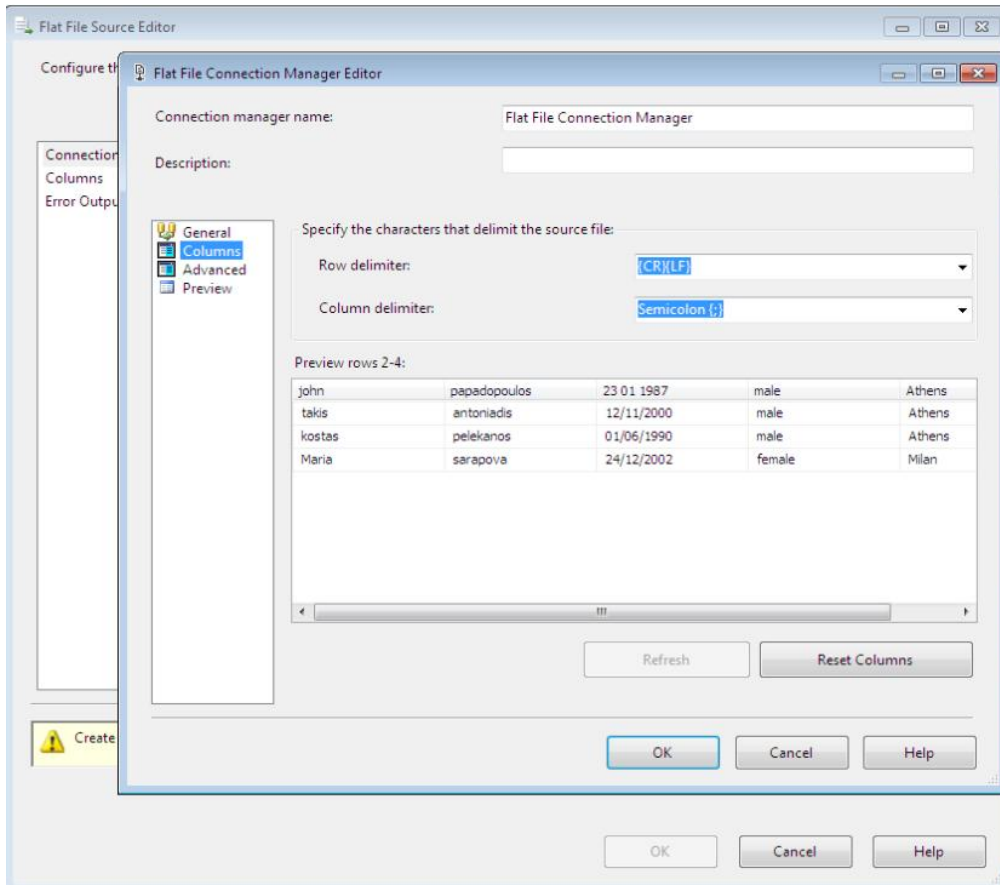
Στην εικόνα 27 στο Flat File Source Editor και στο tab Connection Manager επιλέγουμε ένα new connection για να συνδέσουμε το flat file source με το Customers.txt. Στο κάτω μέρος της οθόνης βλέπουμε ότι το wizard μας εμφανίζει διάφορα tips για να δημιουργήσουμε το connection manager κάνοντας click στο κουμπί new. Αν υπάρχουν και θέλουμε να διατηρήσει τις null τιμές κάνουμε check στο checkbox «Retain null values from the source as null values in the data flow» διαφορετικά θα τις αγνοήσει.



Εικόνα 28 "Browse του αρχείου Customers.txt"

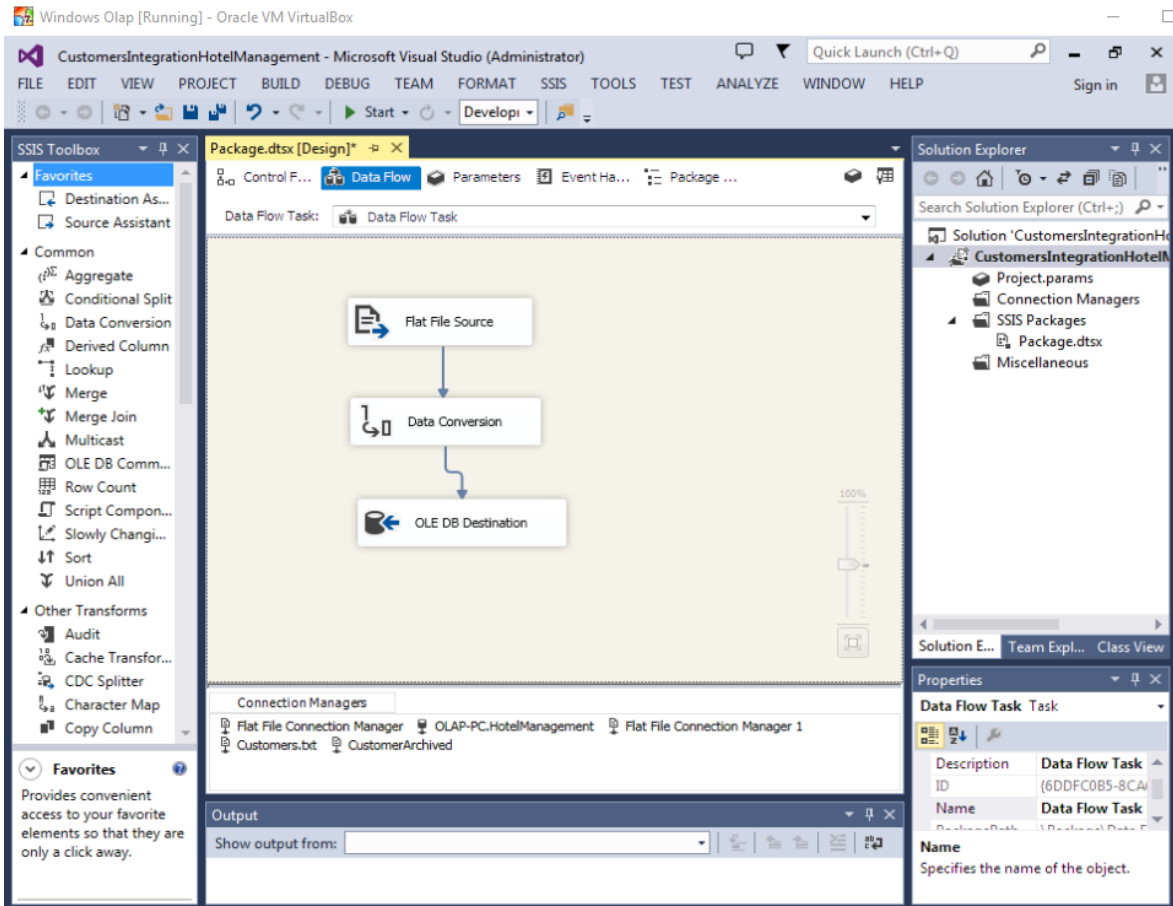
Στην επόμενη οθόνη στο κουμπί Browse του Flat File Connection Manager Editor βρίσκουμε το αρχείο που μας ενδιαφέρει στο δίσκο. Αφού το επιλέξουμε παρατηρούμε ότι ο wizard διαβάζει και αναγνωρίζει το αρχείο και το format του.

Στο Header rows to skip μπορούμε να επιλέξουμε πόσες γραμμές από το πάνω μέρος του αρχείου θα εξαιρέσουμε από τα δεδομένα. Το check box «Column names in the first data row» το αφήνουμε checked αν στο πάνω μέρος του αρχείου έχουμε ονομασίες των στηλών, στην περίπτωση μας το κάνουμε uncheck. Στο format του αρχείου βλέπουμε ότι έχει αναγνωρίσει και έχει προεπιλέξει delimited.



Εικόνα 29 "Flat File Columns"

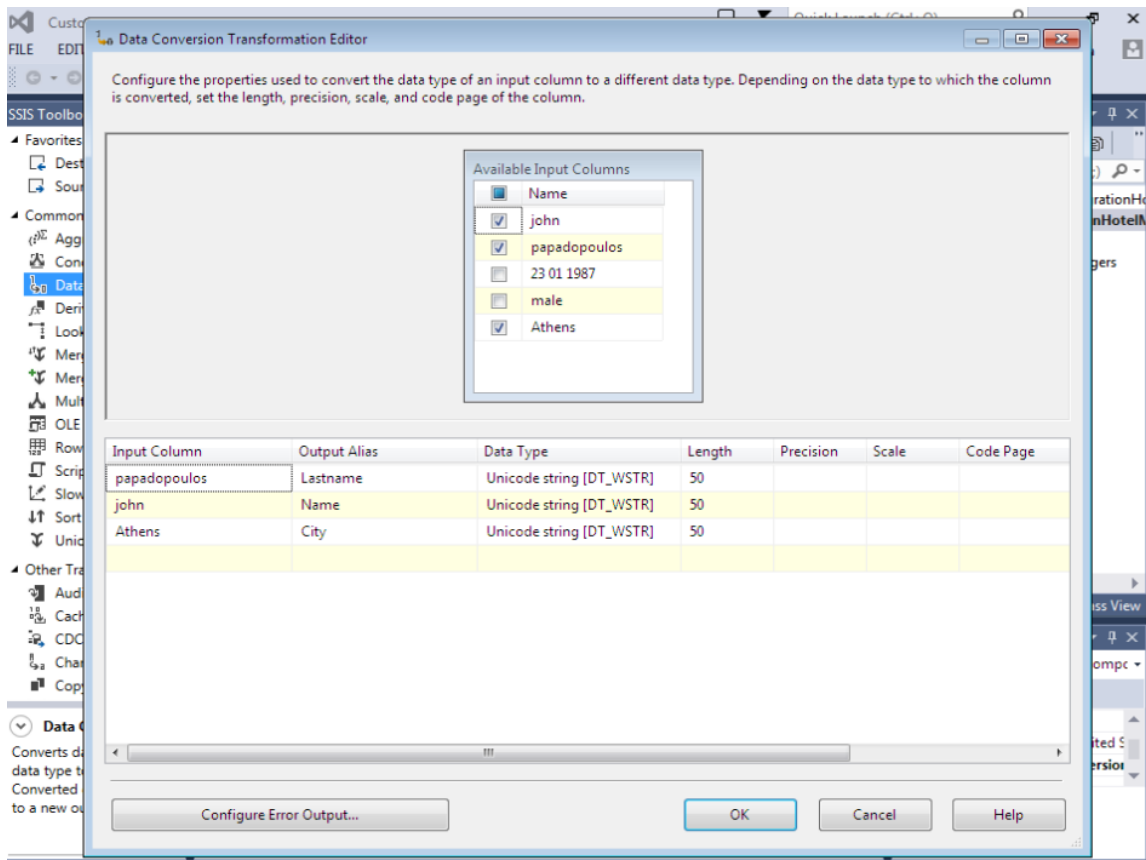
Στην εικόνα 29 και στο tab columns παρατηρούμε ότι ο wizard έχει διαβάσει και έχει αναλύσει το αρχείο με βάση τον delimiter που έχει το αρχείο. Στο preview βλέπουμε ότι έχει χωρίσει σε στήλες τα δεδομένα με βάση τον delimiter “;” και σε νέες γραμμές με βάση το “CR”, “LF” δηλαδή την εισαγωγή νέας γραμμής με την χρήση “enter”.



Εικόνα 30 "Data Conversion and OLE DB Destination"

Στο επόμενο βήμα κάνουμε drag and drop από το toolbox ένα «Data Conversion» και ένα «OLE DB Destination» όπως βλέπουμε στην εικόνα 30. Συνδέουμε το «flat file Source» με το «Data Conversion». Το «Data Conversion» έχει σαν σκοπό να μετατρέψει τα δεδομένα από 1252 ANSI που ήταν στο txt αρχείο σε μια μορφή που μπορεί να αναγνωρίσει ο SQL Server. Στην περίπτωση μας κάνοντας διπλό click στο Data Conversion μας ανοίγει η οθόνη που βλέπουμε στην εικόνα 31.

Να σημειώσουμε σε αυτό το σημείο ότι καθώς δημιουργούμε elements τα οποία αφορούν κάποια σύνδεση του project σε ένα αρχείο ή σε μια βάση, είτε είναι για να πάρουμε είτε για να στείλουμε αρχεία, το project δημιουργεί τα αντίστοιχα connection managers που διαχειρίζονται την σύνδεση. Μπορούμε να παρατηρήσουμε τα Connection Manager που έχουν δημιουργηθεί στο κάτω μέρος στην εικόνα 30. Τα connections που υπάρχουν είναι του αρχείου txt για μεταφορά των δεδομένων στο data flow, του αρχείου txt σαν file type για την μεταφορά του στο δίσκο και του folder Customer Archived για την μεταφορά του αρχείου στο δίσκο μέσα στο File System Task. Επίσης τα connections της βάσης δεδομένων που θα φορτωθούν τα δεδομένα του OLE DB Destination.

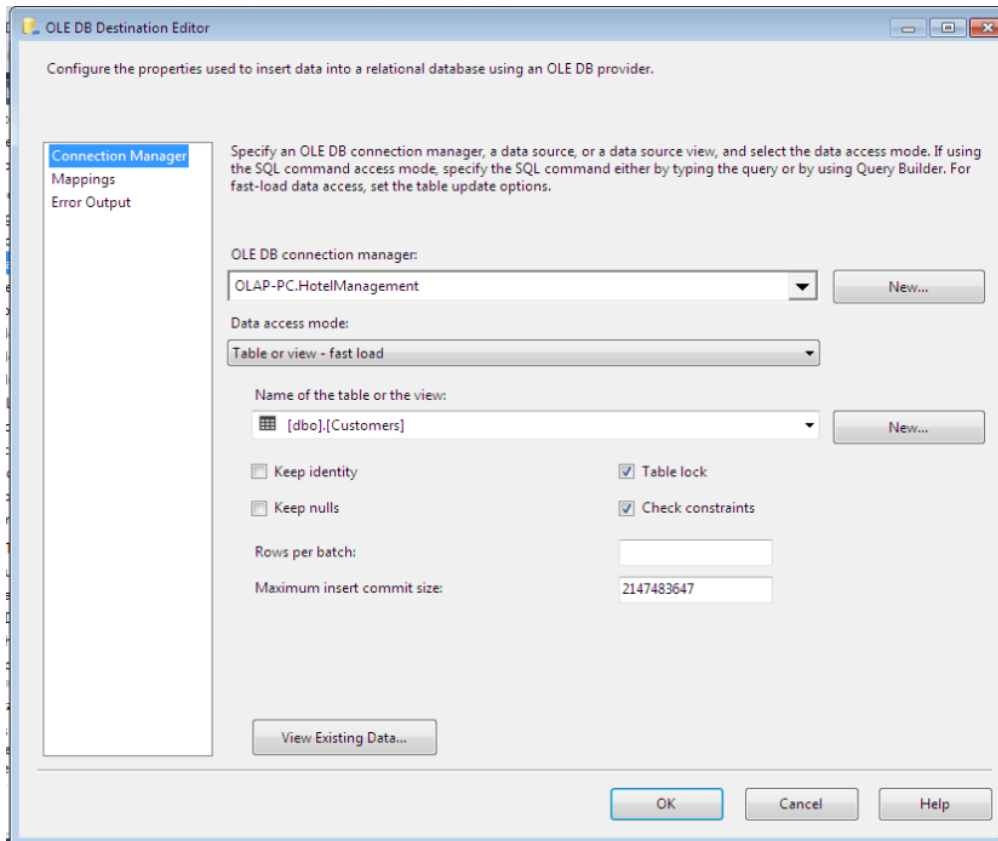


Εικόνα 31 "Data Conversion and Data Mapping"

Σε αυτό το σημείο επιλέγουμε πια δεδομένα από το flat file source θέλουμε να κρατήσουμε. Ύστερα ορίζουμε το «output Alias» για να δώσουμε ένα όνομα στις στήλες των δεδομένων. Στο «Data type» επιλέγουμε Unicode String[DT\_WSTR] όπου είναι μια μορφή που μπορεί να χειριστεί το OLE DB Destination για να περάσει τα δεδομένα στον πίνακα Customers. Τέλος επιλέγουμε το length φροντίζοντας να καλύπτει το μέγεθος των δεδομένων. Αν το length που επιλέξουμε είναι μικρότερο από αυτό που είναι κάποιο στοιχείο η εφαρμογή θα «κόψει» την λέξη στο τέλος για να συμμορφωθεί με τον κανόνα.

Κλείνοντας τις λειτουργίες του Data Flow πρέπει να κάνουμε διπλό click στο «OLE DB Destination» και να ορίσουμε την βάση δεδομένων που θα συνδεθούμε και τον πίνακα που θα βάλουμε τα δεδομένα.

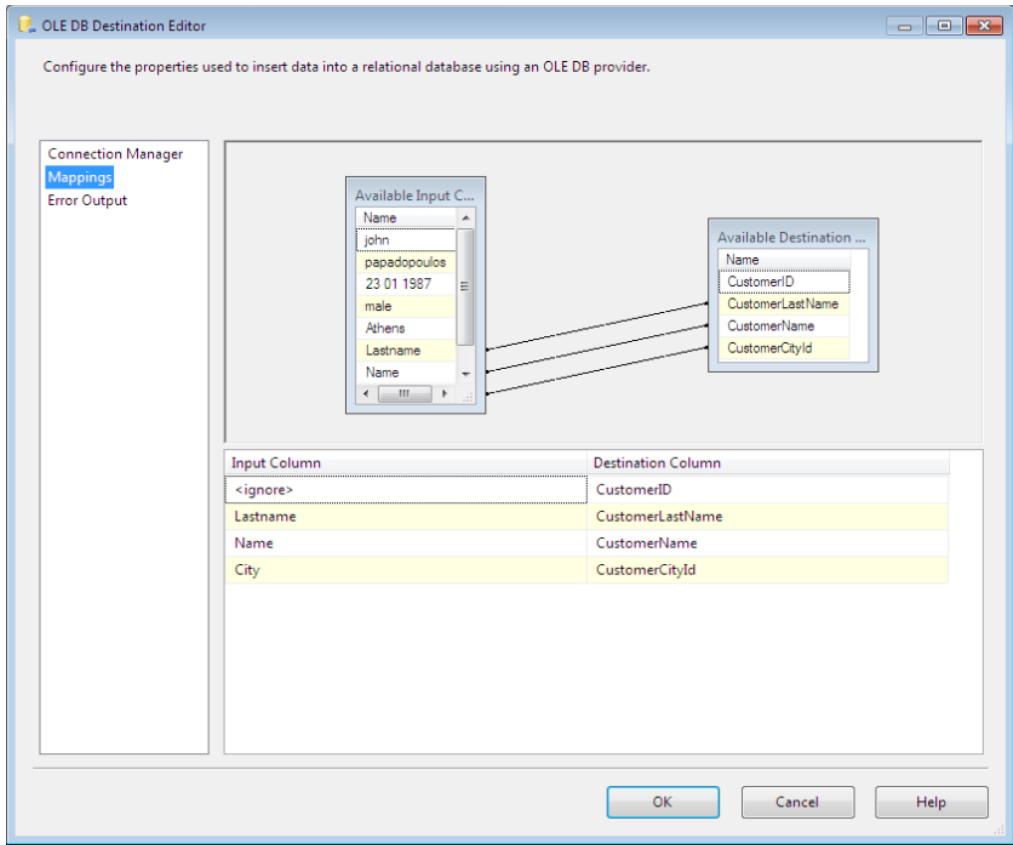




Εικόνα 32 "OLE DB Destination Connection Manager"

Στην εικόνα 32 βλέπουμε ότι στο tab Connection Manager έχουμε επιλέξει την βάση OLAP-PC.HotelManagement και τον πίνακα Customers.

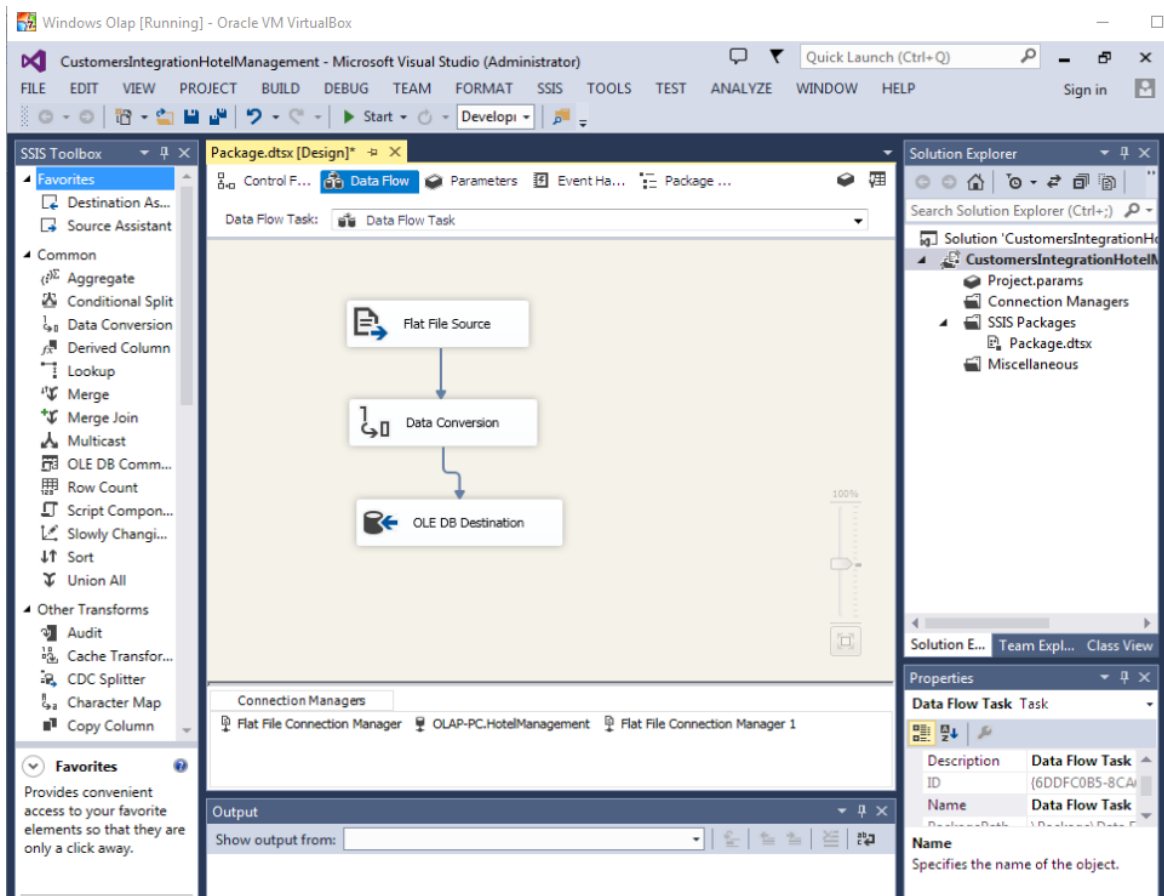
Το επόμενο βήμα είναι να συνδέσουμε τις στήλες που παράχθηκαν από το Data Conversion με τις στήλες του πίνακα Customers. Το maximum insert commit size μας δείχνει τα όρια του SSIS σε αυτή τη μεταφορά των δεδομένων. Επίσης βλέπουμε ότι έχουμε κάποιες επιλογές όπως να κλειδώσουμε τον πίνακα κατά την διάρκεια εισαγωγής ώστε να μην μπορεί να τον χρησιμοποιήσει κάποια άλλη τρίτη εφαρμογή. Να κρατήσουμε το identity των κελιών. Να κρατήσουμε τις null εγγραφές σε περίπτωση που υπάρχουν όπως επίσης να υπολογίσει και τα constraint του πίνακα σε περίπτωση που μας ενδιαφέρουν.



Εικόνα 33 "OLE DB Destination Mappings"

Στην εικόνα 33 στο «OLE DB Destination Editor» έχουμε συνδέσει τις στήλες από το "input Column" με τις στήλες στο "Destination Column". Το wizard οπτικοποιεί την σύνδεση των δεδομένων μεταξύ των εικονικών πινάκων και συνδέει τις στήλες μεταξύ τους. Τέλος πατάμε «οκ» και επιστρέφουμε στο Design του Data Flow. Σε αυτό το σημείο το Data Flow έχει ολοκληρωθεί.

Στο tab error output μπορούμε να του ορίσουμε τι θα κάνει σε περίπτωση failure όπως αν θα σταματήσει το project, αν θα αγνοήσει το failure ή αν θα κάνει redirect το row σε κάποιο άλλο εργαλείο για να προσπαθήσει να το χειριστεί.

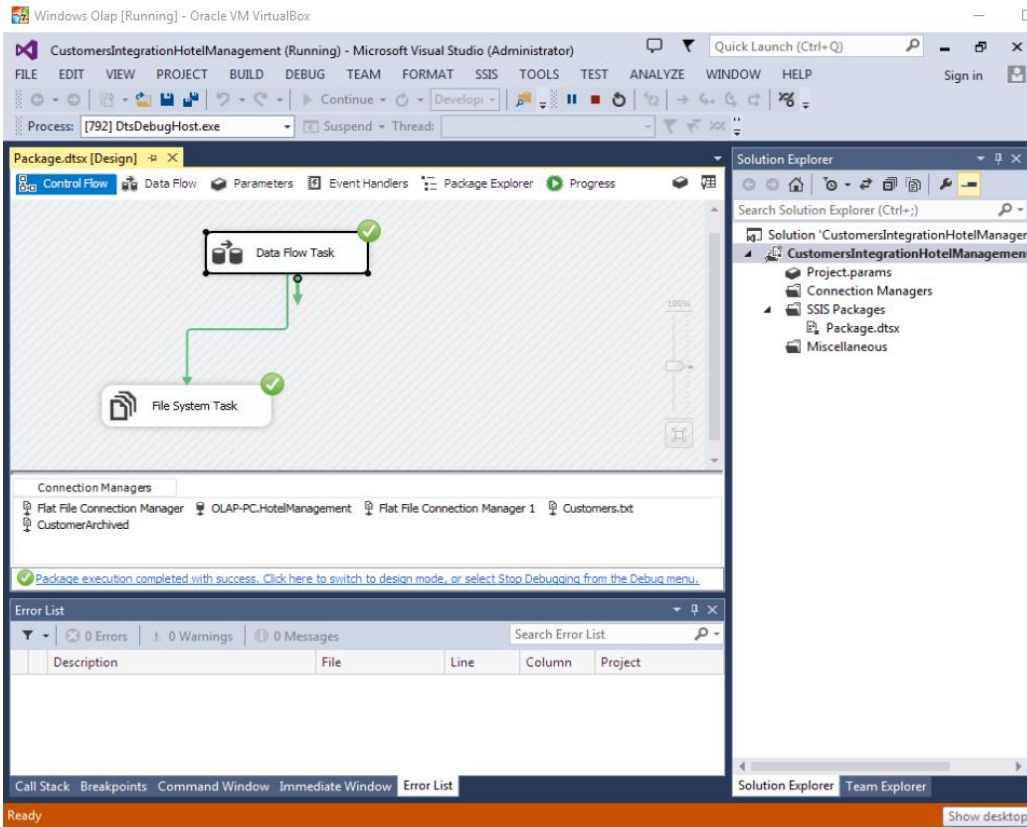


Εικόνα 34 "Data Flow"

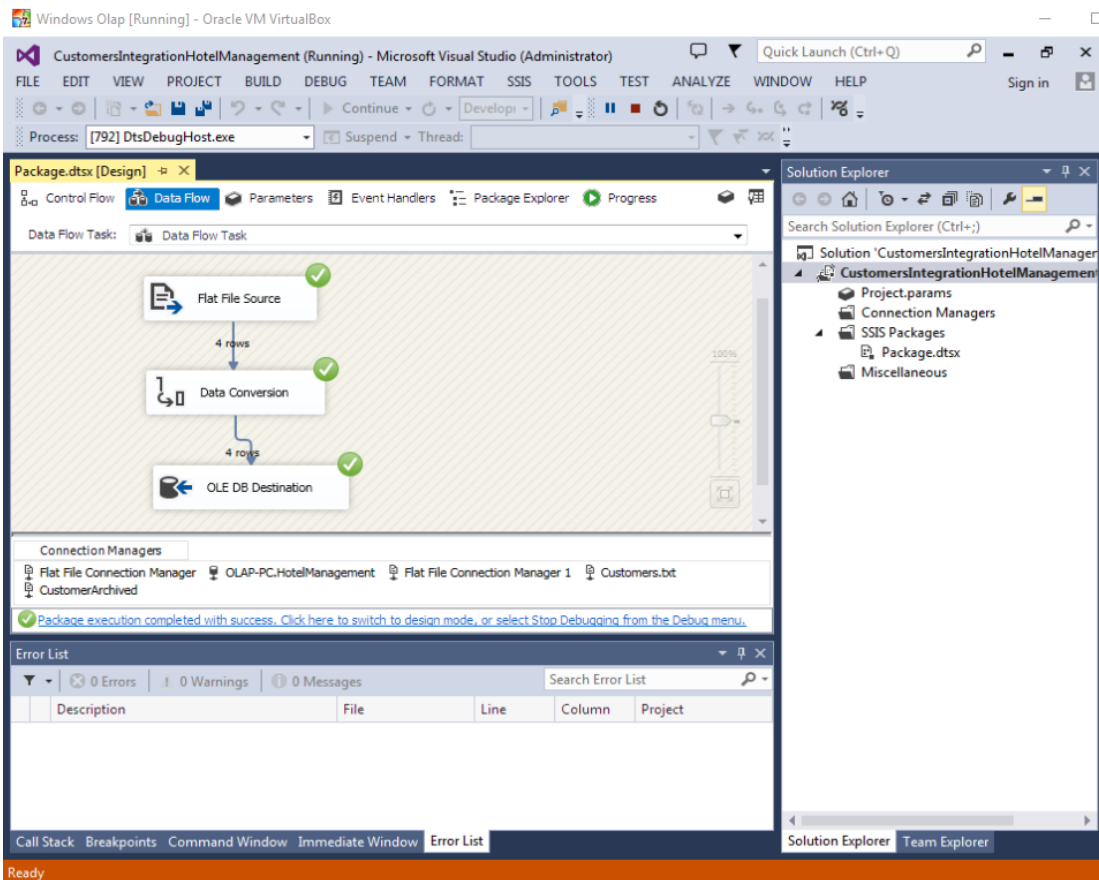
Κατά την εκτέλεση του SSIS και συγκεκριμένα του Data Flow task θα εκτελέσει σειριακά όλα τα στοιχεία του Data Flow και όταν εκτελέσει και το τελευταίο OLE DB Destination επιτυχώς, θα γυρίσει πίσω στο Control Flow, και θα ολοκληρωθεί επιτυχώς η εκτέλεση του Data Flow Task. Σε αυτό το σημείο θα έχουν φορτωθεί τα δεδομένα στην βάση και θα πάει στο επόμενο βήμα για να εκτελέσει το File System Task και να μεταφέρει το αρχείο Customers.txt από το desktop στο folder Customers Archived. Μόλις ολοκληρωθεί και το File System Task επιτυχώς θα ολοκληρωθεί το project.

Στις επόμενες δύο εικόνες μπορούμε να δούμε τα αποτελέσματα του τρεξίματος της εφαρμογής και συγκεκριμένα του Control Flow και του Data Flow αφού πατήσουμε start στο project.

## Πτυχιακή εργασία του φοιτητή Δουλάμη Δημήτρη



Εικόνα 35 "Control Flow Run"

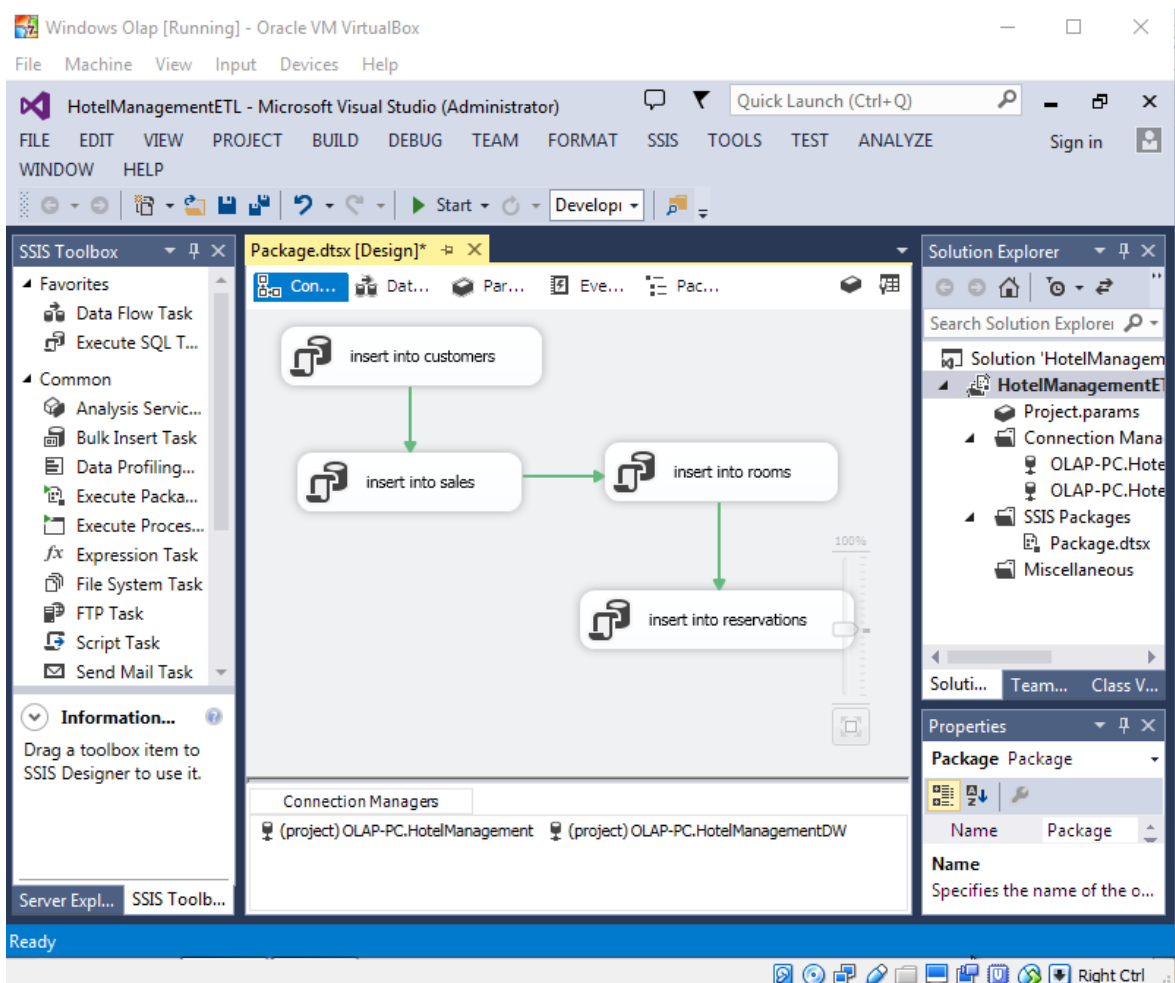


Εικόνα 36 "Data Flow Run"

Σε αυτό το σημείο έχουμε ολοκληρώσει την διαδικασία εισαγωγής στην βάση το καινούριο πελατολόγιο του ταξιδιωτικού πράκτορα μέσω ενός πακέτου SSIS και παρουσιάσαμε αναλυτικά το Control Flow και το Data flow του Design μαζί με κάποια χρήσιμα εργαλεία.

Στην εφαρμογή Hotel Management ETL της μεταφοράς δεδομένων από την βάση στην αποθήκη προσεγγίζουμε ένα διαφορετικό τρόπο αντιμετώπισης της διαδικασίας μεταφοράς δεδομένων. Χρησιμοποιούμε την γλώσσα SQL σε Execute SQL tasks στο Control Flow για να μεταφέρουμε τα δεδομένα από την μια βάση στην άλλη.

Όπως φαίνεται στην εικόνα 37 αφού δημιουργήσαμε τα connection managers εισάγουμε από το SSIS toolbox το execute SQL task στο οποίο θα γράψουμε τον SQL κώδικα για την μεταφορά των δεδομένων. Στην περίπτωση μας η βάση έχει τέσσερις πίνακες συνεπώς θα χρειαστεί να κάνουμε drag and drop στο control flow τέσσερα διαφορετικά Execute SQL Tasks.

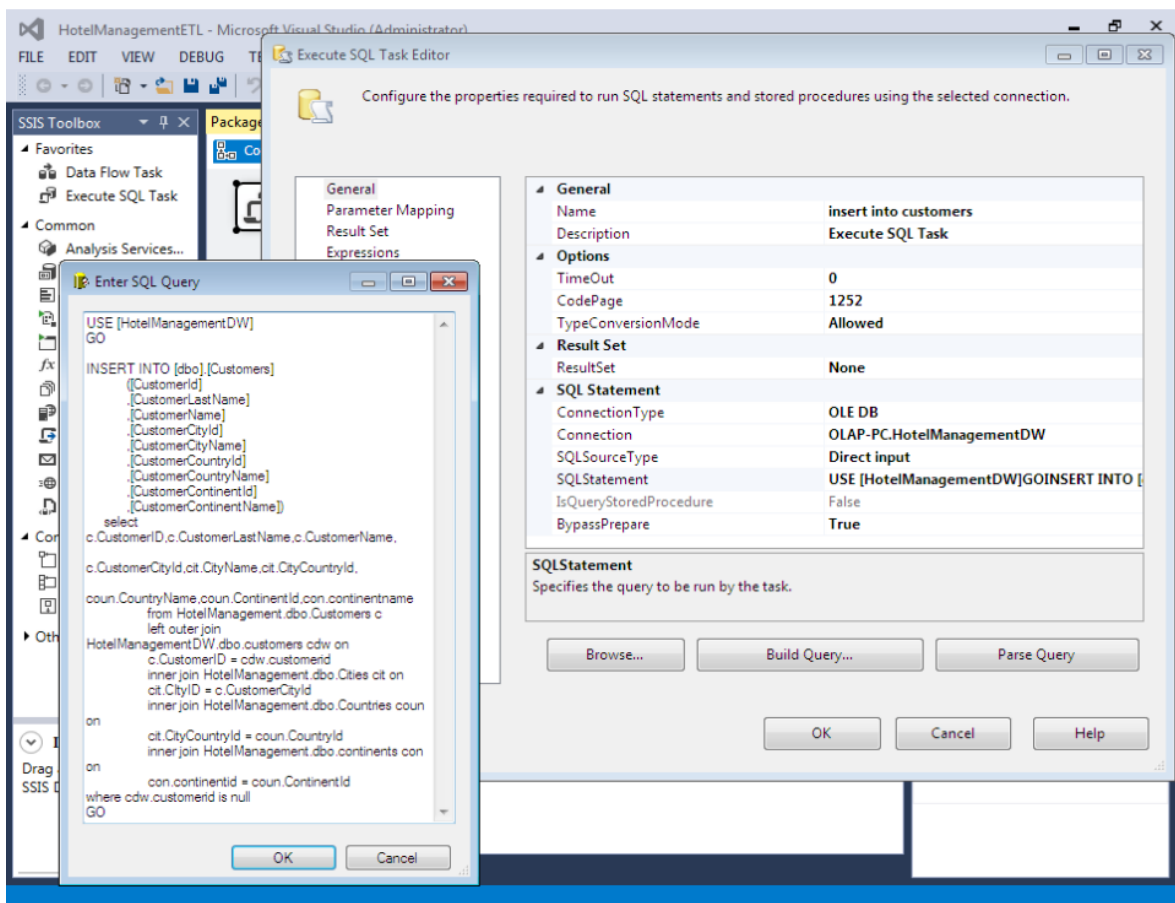


Εικόνα 37 "Control Flow"

Το κάθε ένα αναλαμβάνει την εισαγωγή σε ένα πίνακα. Στην εικόνα 37 βλέπουμε πως θα είναι τελικά το control flow του SSIS. Στα SQL scripts αυτά πρέπει να υλοποιήσουμε την λογική της εισαγωγής των δεδομένων στην αποθήκη. Την πρώτη φορά που το τρέχουμε περνάει όλα τα δεδομένα από την βάση στην αποθήκη. Την δεύτερη φορά θα πρέπει να βάλουμε ένα έλεγχο ώστε να αποφύγουμε το να ξαναπεράσει δεδομένα που ήδη έχουν περαστεί, διπλές εγγραφές και επίσης να αποφύγουμε σφάλματα primary key violation.

Αυτό που πρέπει να κάνουμε είναι ένα select SQL query που θα φέρνει τις εγγραφές που θέλουμε κάνοντας τα αντίστοιχα inner joins μεταξύ των πινάκων.

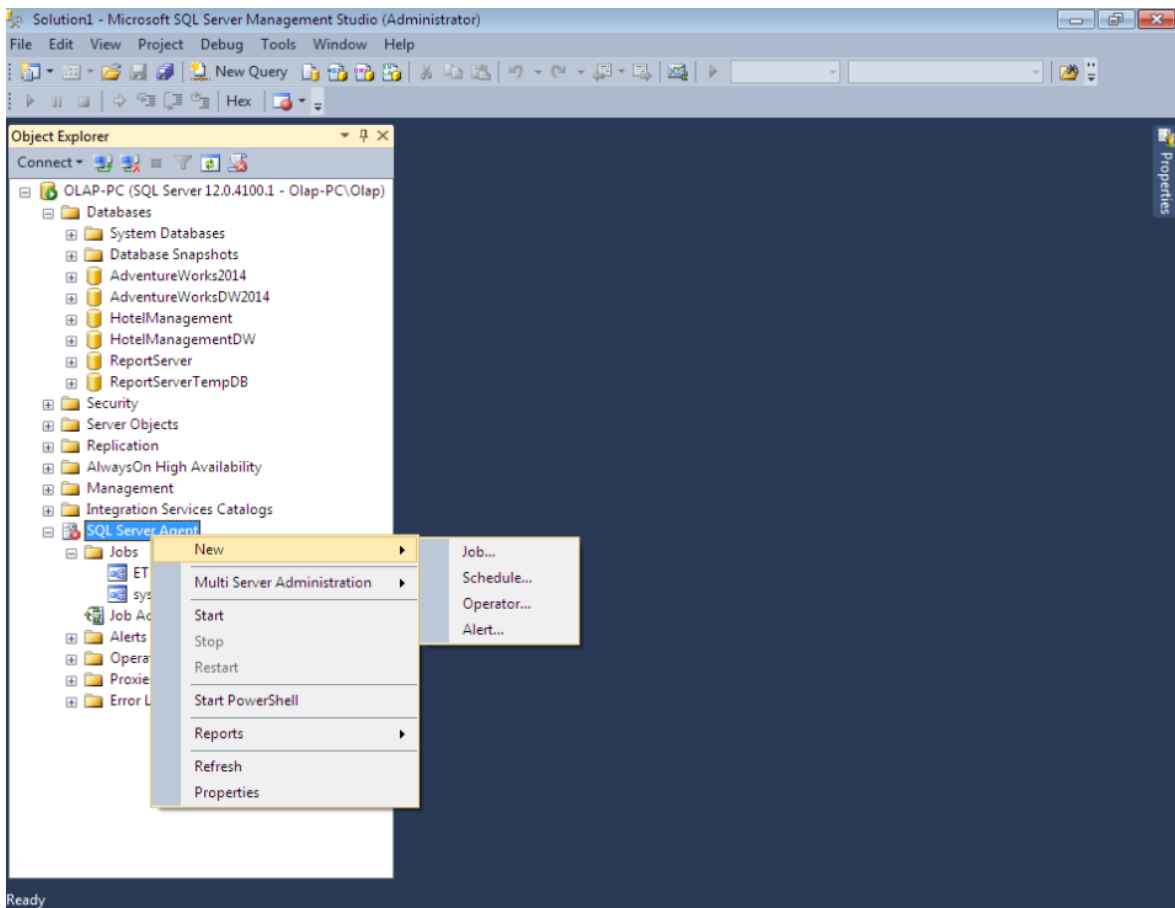
Ο τρόπος που χρησιμοποιούμε για να αποφύγουμε τις διπλές εγγραφές είναι να κάνουμε **left outer join** τους δύο πίνακες, που κρατάνε τα σχετικά στοιχεία, μεταξύ τους και να επιλέξουμε μόνο τις εγγραφές που δεν υπάρχουν ήδη στο πίνακα της αποθήκης δεδομένων, με την χρήση «**where "field" is null**». Ακολουθούμε την ίδια τεχνική και για τους υπόλοιπους πίνακες. Στην εικόνα 38 βλέπουμε το SQL Task καθώς και το Select statement.



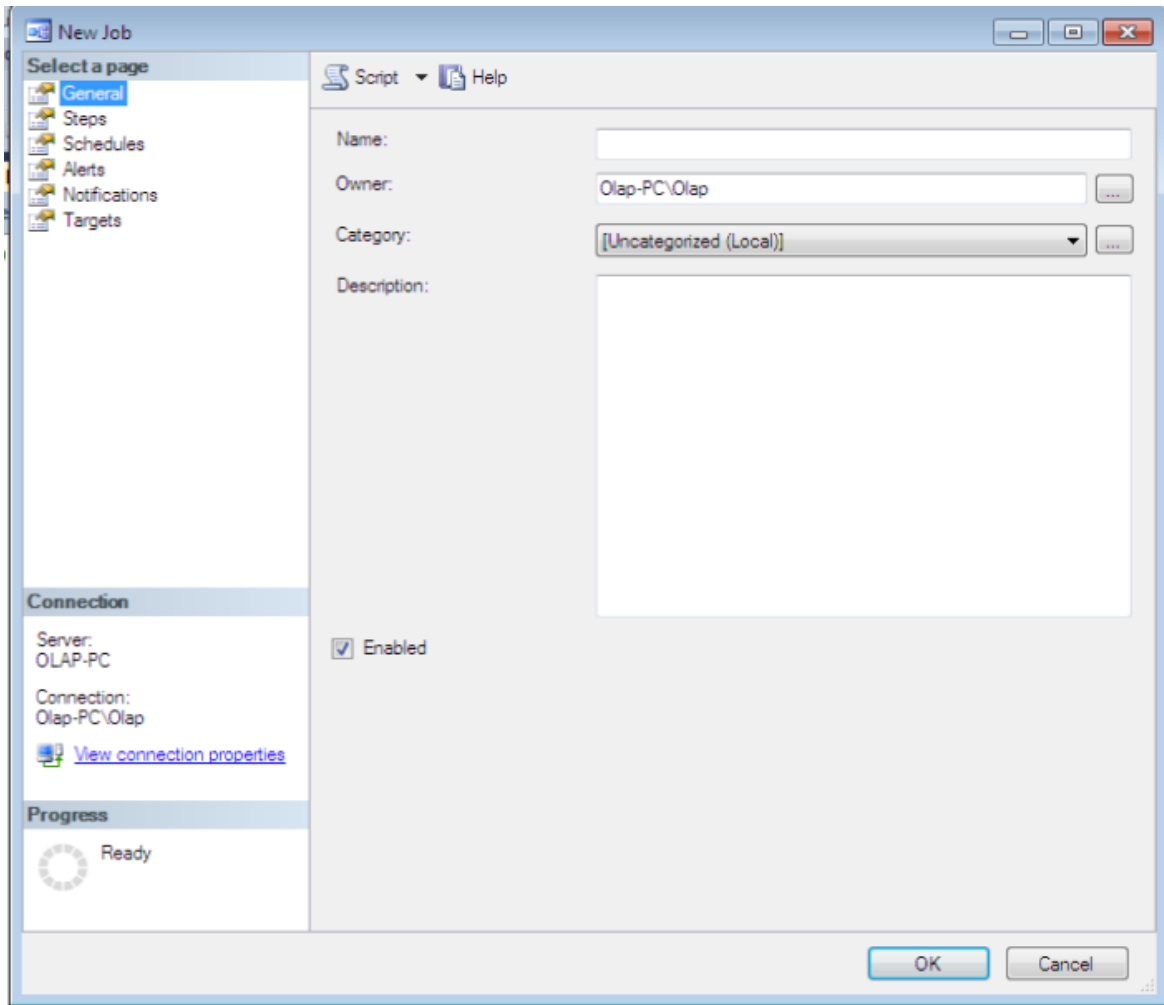
Εικόνα 38 "Execute SQL Task"

Με αυτόν τον τρόπο όταν με την πάροδο του χρόνου τα δεδομένα στην κύρια βάση θα έχουν αυξηθεί τρέχουμε το SSIS πακέτο και γεμίζει την αποθήκη μόνο με τα καινούρια δεδομένα.

Για να μην τρέχουμε το SSIS πακέτο μέσα από το visual studio κάθε φορά που το χρειαζόμαστε, μπορούμε να το κάνουμε import στον SQL server σαν ένα νέο πακέτο. Αυτή την διαδικασία την προσφέρει το Management Studio μέσω του SQL Server Agent και αυτοματοποιεί πλήρως την εκτέλεση SSIS πακέτων. Μέσα στον Server Agent μπορούμε να δημιουργήσουμε αυτοματοποιημένες διαδικασίες για διάφορες λειτουργίες. Στον object explorer στο μενού SQL server agent δημιουργούμε ένα νέο «job», επιλέγουμε SQL Server integration και εισάγουμε το αρχείο dtsx που βρίσκεται στο folder του visual studio project.



Εικόνα 39 "New SQL Server Agent Job"



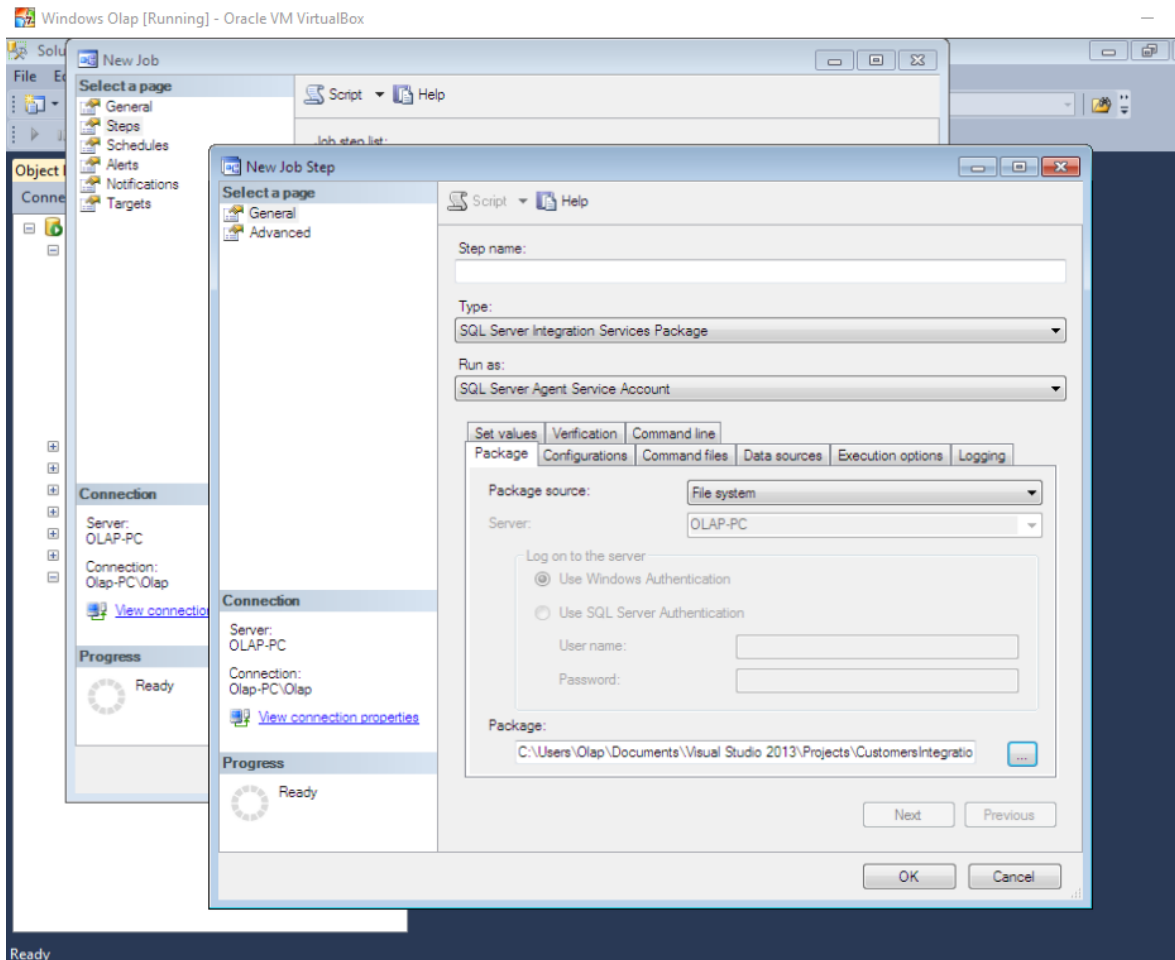
Εικόνα 40 "New Job"

Αναλυτικότερα στην εικόνα 40 βλέπουμε τα διάφορα tabs στην διαδικασία δημιουργίας ενός καινούριου job στο Management Studio. Στο tab General ορίζουμε όνομα για το job, τον Owner που θα είναι ο χρήστης του συστήματος OLAP και γράφουμε ένα description για documentation του job.

Στο tab Steps πατάμε το κουμπί new για να δημιουργήσουμε ένα καινούριο βήμα στο job.

Μας ανοίγει το panel «New Job Step» που φαίνεται στην εικόνα 41. Το πρώτο πράγμα που πρέπει να κάνουμε είναι να δώσουμε ένα όνομα στο step. Στο dropdown «Type» επιλέγουμε το SQL Server Integration Services Packages για να δηλώσουμε ότι θα πρέπει να εκτελέσει ένα πακέτο SSIS. Στο run as επιλέγουμε τον SQL Server Agent Service Account. Τέλος στο tab «package» επιλέγουμε σαν package source το «file system» γιατί θα του δώσουμε το dtsx αρχείο από το project του Visual Studio. Στο package βρίσκουμε το αρχείο dtsx που μας ενδιαφέρει και βρίσκεται μέσα στο φάκελο του project που «χτίσαμε» προηγουμένως.





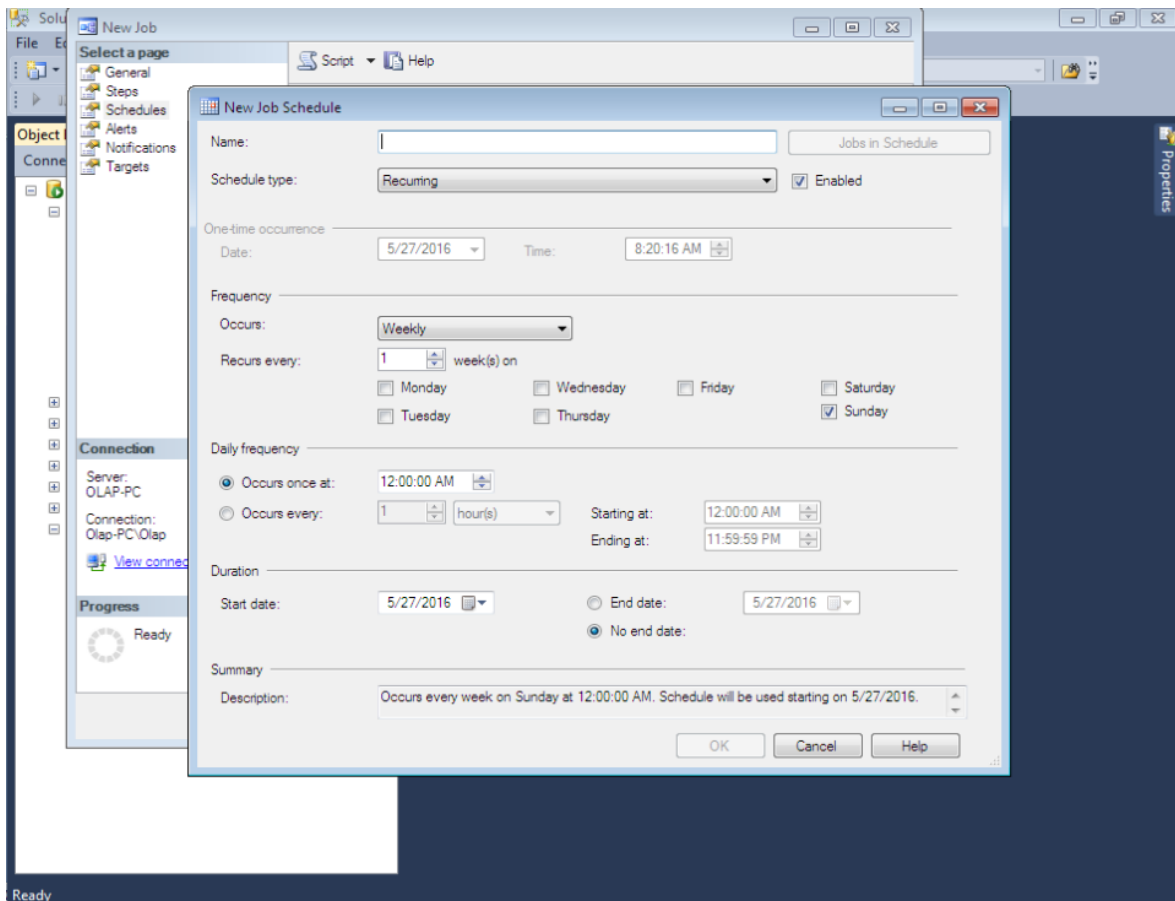
Εικόνα 41 "New Job Step"

Στο τέλος πατάμε ok για να δημιουργηθεί το job. Το καινούριο job θα μπει στον φάκελο jobs στον solution explorer κάτω από τον SQL Server Agent και για να το εκτελέσουμε πατάμε δεξί click και start job at step.

Σε αυτό το σημείο πρέπει να υπογραμμίσουμε ότι ο SQL Server Agent είναι ένα windows service και πρέπει να φροντίσουμε να τρέχει σε περίπτωση που είναι σταματημένος για κάποιο λόγο. Εκτός από SSIS πακέτα ο SQL Server Agent μπορεί να αυτοματοποιήσει διάφορες λειτουργίες του Management Studio όπως SQL Scripts, SSAS Scripts, powershell, cmdExec αρχεία, replications, logs και διάφορα άλλα.

Ο SQL Server Agent τέλος μας παρέχει την δυνατότητα scheduling. Αυτό μας επιτρέπει να ορίζουμε ένα job να τρέχει σε ημερομηνίες της επιλογής μας με ένα τρόπο πλήρως αυτοματοποιημένο. Για παράδειγμα μπορούμε να δημιουργήσουμε ένα job που παίρνει backup την βάση και να το ορίσουμε να τρέχει κάθε βράδυ στις 12:00 ή μια φορά την εβδομάδα την Κυριακή στις 12:00 το βράδυ που ο φόρτος είναι ο ελάχιστος. Στην εικόνα 42 βλέπουμε την οθόνη αφού έχουμε πατήσει ένα new schedule στο Tab Schedules. Στην αρχή δίνουμε ένα όνομα στο

schedule. Επιλέγουμε στο type αν θέλουμε να είναι Recurring δηλαδή να τρέχει ανά διαστήματα ή αν θέλουμε να τρέξει μια φορά. Στην περίπτωση του recurring ορίζουμε συχνότητα βάση μέρας, βδομάδας ή μήνα. Στο summary μας ενημερώνει κάθε πότε το προγραμματίσαμε να εκτελείται και πότε είναι η επόμενη φορά που θα εκτελεστεί.

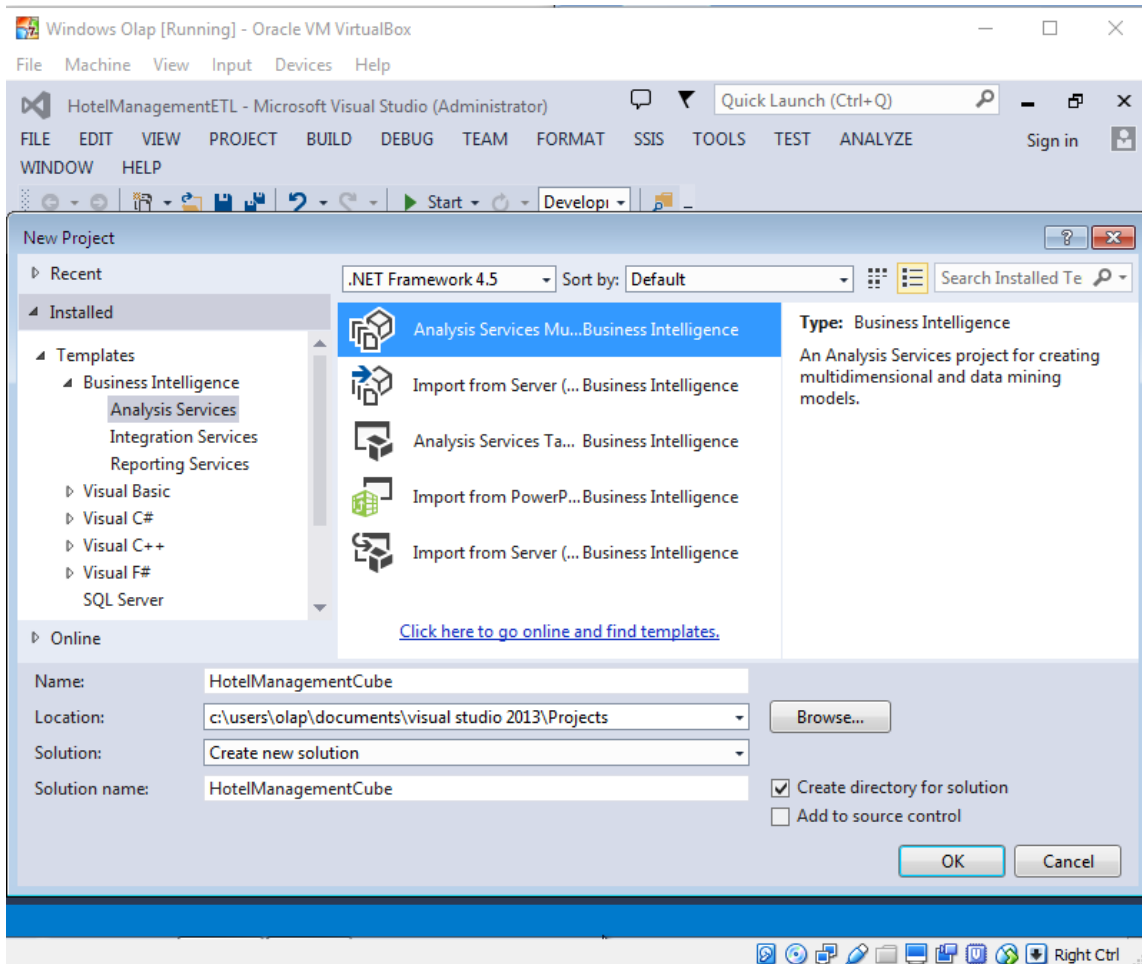


Εικόνα 42 "SQL Server Agent Job Schedules"

### 3.6 Ο κύβος OLAP Hotel Management Cube

Σε αυτό το κεφάλαιο θα δούμε αναλυτικά την δημιουργία του κύβου και των διαστάσεων. Στο visual studio θα δημιουργήσουμε ένα νέο Analysis Services project και μέσα θα δημιουργήσουμε τον κύβο με τα δεδομένα της βάσης Hotel Management. Οι επιλογές που έχουμε στα templates στο πακέτο Business Intelligence του Visual Studio είναι Analysis Services, Integration Services και Reporting Services. Στο Integration Services δημιουργήσαμε το SSIS πακέτο για χρησιμοποιείται κυρίως για αυτοματοποιημένες εργασίες πάνω στην βάση δεδομένων. Στο Reporting Services δεν θα επεκταθούμε σε αυτή την πτυχιακή χρησιμοποιείται για την παραγωγή reports από την βάση δεδομένων και ο SQL

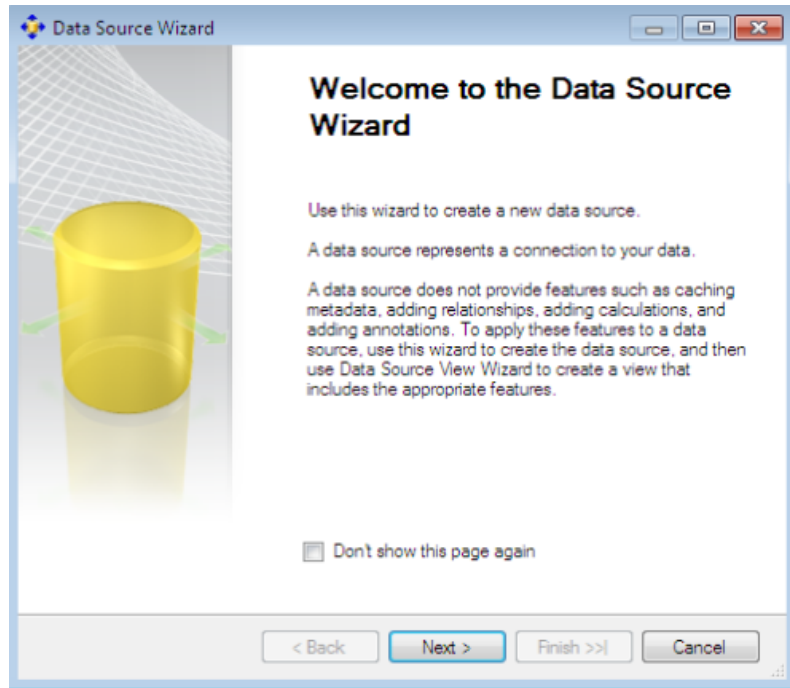
Server μας παρέχει και τον Report Server που μπορεί να στεγάσει αυτά τα reports για να έχουν πρόσβαση σε αυτά και άλλοι χρήστες μια επιχείρησης. Analysis Services είναι η τεχνολογία OLAP και πάνω σε αυτό θα δημιουργήσουμε τα project μας.



Εικόνα 43 "Νέο Project Analysis Services"

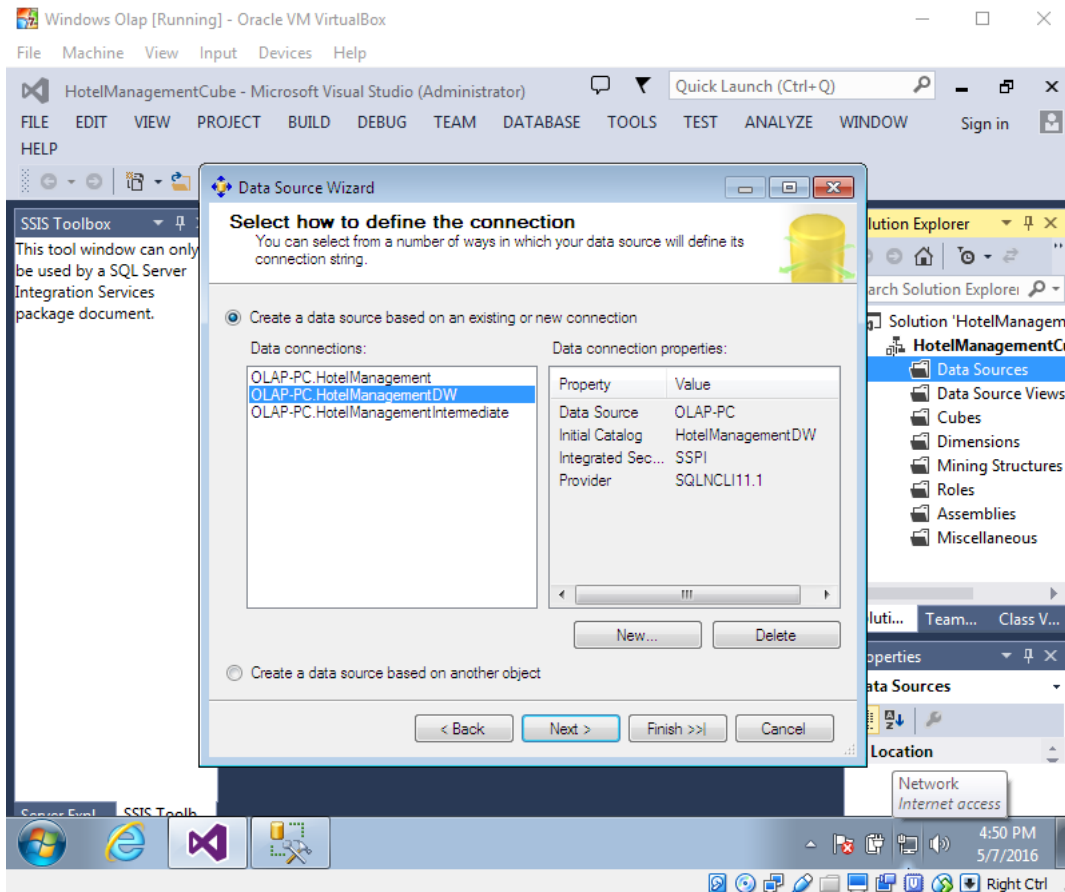
Το πρώτο πράγμα που πρέπει να κάνουμε αφού δημιουργήσουμε ένα νέο Analysis Services multidimensional and data mining project είναι να ορίσουμε ένα data source για να πάρουμε τα δεδομένα. Το data source αντιπροσωπεύει την σύνδεση με τα δεδομένα.

Το data Source θα είναι η πηγή δεδομένων για το Data Source View. Στο πεδίο «data source» επιλέγουμε «new data source» και μας ανοίγει το wizard δημιουργίας data source.



Εικόνα 44 "Data Source Wizard"

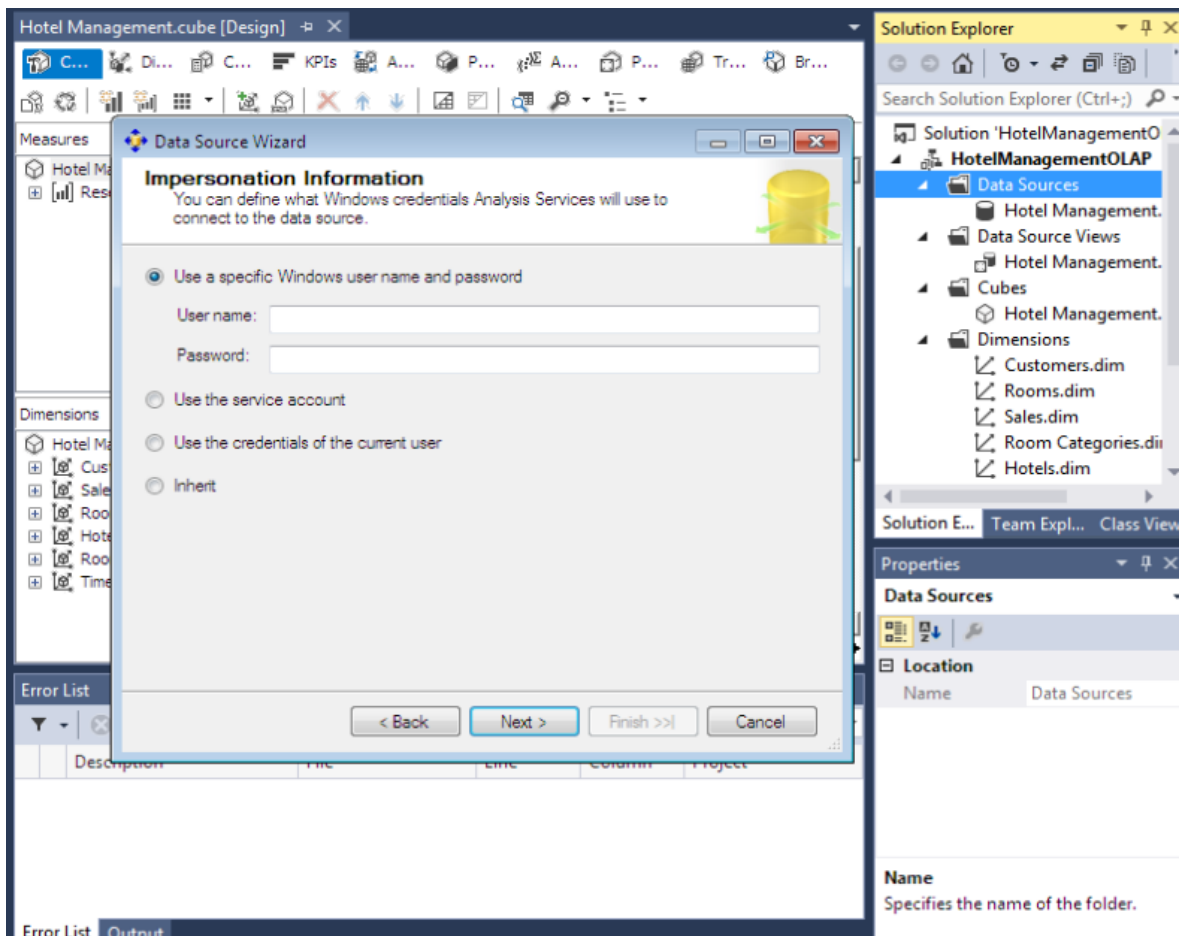
Πατάμε next και στην επόμενη οθόνη επιλέγουμε «new connection». Στο data source ορίζουμε την πηγή των δεδομένων που θα χρησιμοποιήσουμε για να φτιάξουμε τον κύβο. Θα δημιουργήσουμε το connection στον SQL Server που έχουμε την βάση δεδομένων και θα επιλέξουμε την βάση που θέλουμε για να κάνουμε connect.



Εικόνα 45 "Νέο data source - connection Hotel Management DW"

Στο επόμενο βήμα ορίζουμε τα Impersonation information όπως φαίνεται στην επόμενη σελίδα στην εικόνα 46.

Μπορούμε να επιλέξουμε «Use Specific Windows user name and password» και να βάλουμε τα credentials του χρήστη. Με το «Use the credentials of the current user» χρησιμοποιεί τα credentials του τωρινού χρήστη. Με το «inherit» κληρονομεί τα credentials από το connection της βάσης. Τέλος πατάμε next. Στην επόμενη οθόνη επιλέγουμε ένα όνομα για το Data Source και πατάμε finish.



Εικόνα 46 "Impersonation information"

Το δεύτερο βήμα είναι να δημιουργήσουμε ένα data source view. Το Data Source View είναι η όψη των δεδομένων με τους πίνακες και τις συσχετίσεις τους. Στο data source View μπορούμε επίσης να προσθέσουμε ή να πειράξουμε τις συσχετίσεις των πινάκων. Επίσης μπορούμε να αποθηκεύσουμε meta-data και να δημιουργήσουμε calculations.

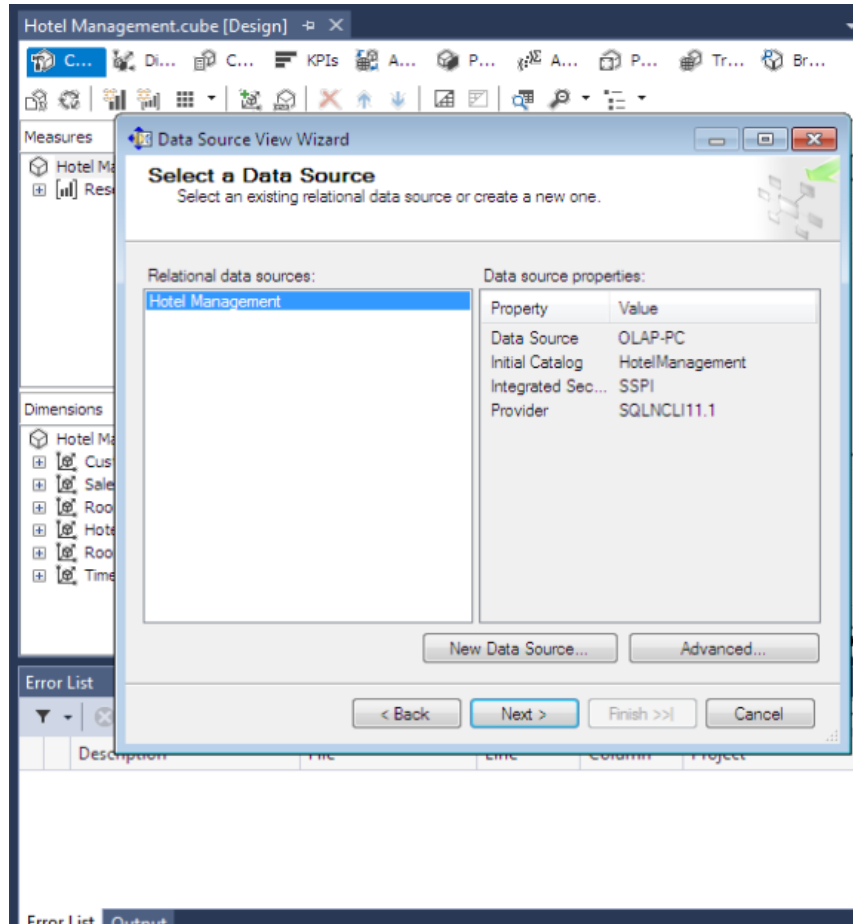
Στο πεδίο data source view επιλέγουμε «new data source view» και μας ανοίγει το wizard δημιουργίας ενός view.



Εικόνα 47 "Data Source view Wizard"

Στην επόμενη οθόνη εικόνα 48 επιλέγουμε το data source που δημιουργήσαμε προηγουμένως πάνω στο οποίο πάνω θα φτιαχτεί το data source view και πατάμε next. Αν θέλουμε να δημιουργήσουμε καινούριο Data Source πατάμε στην επιλογή New Data Source.

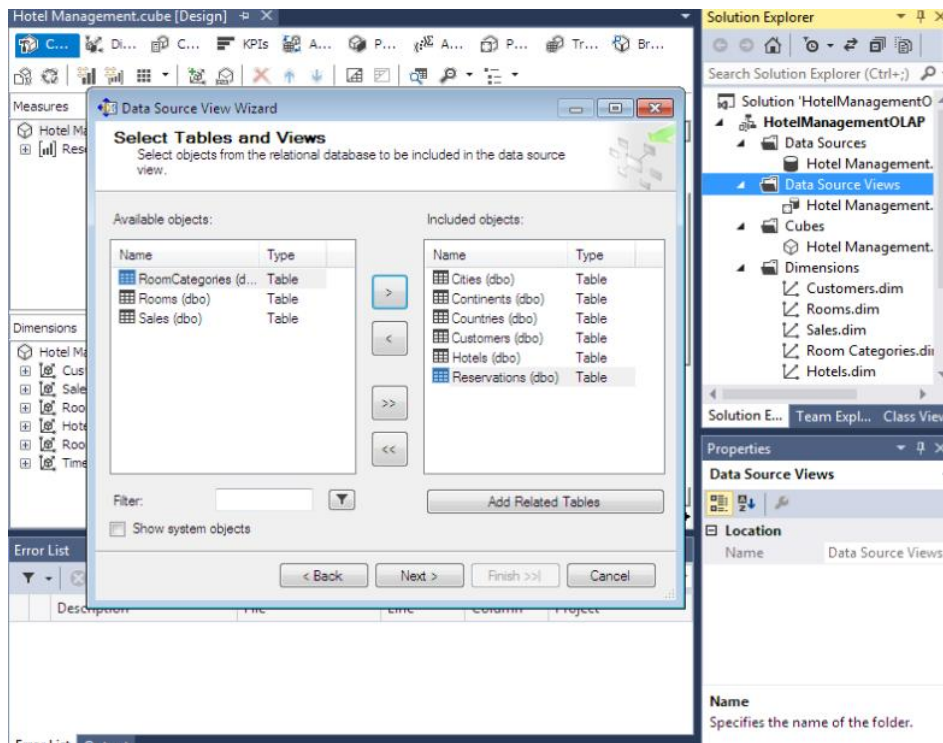
Στο κουμπί advanced ο wizard μας ρωτάει αν θέλουμε να συμπεριλάβει τις σχέσεις μεταξύ των πινάκων το οποίο είναι και η default επιλογή σε κάθε νέο data source View, το αφήνουμε ως έχει. Στο data source properties βλέπουμε κάποιες πληροφορίες, τα ονόματα και τις τιμές τους, που αφορούν το data source.



Εικόνα 48 "Data Source Connection"

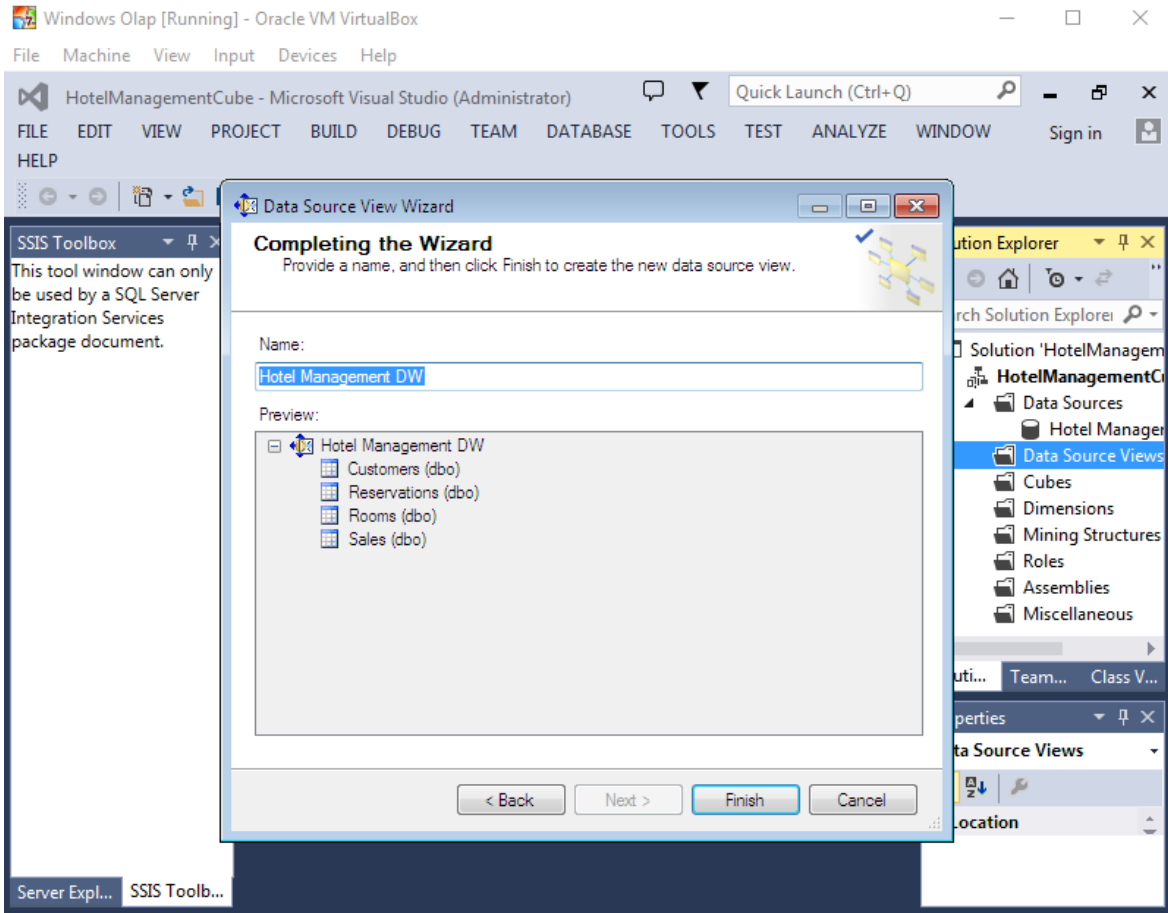
Στην επόμενη εικόνα (εικόνα 49) βλέπουμε την διαδικασία εισαγωγής στο View των πινάκων που επιθυμούμε. Στο available objects το wizard έχει φορτώσει από την βάση τους πίνακες. Στο included objects βρίσκονται οι πίνακες που θα περιέχονται στο View. Επιλέγουμε τους πίνακες που θέλουμε να εισαγάγουμε στο view. Αφού επιλέξουμε τους πίνακες που χρειαζόμαστε πατάμε next και πηγαίνουμε στην τελική οθόνη.





Εικόνα 49 "Επιλογή πινάκων Data Source View"

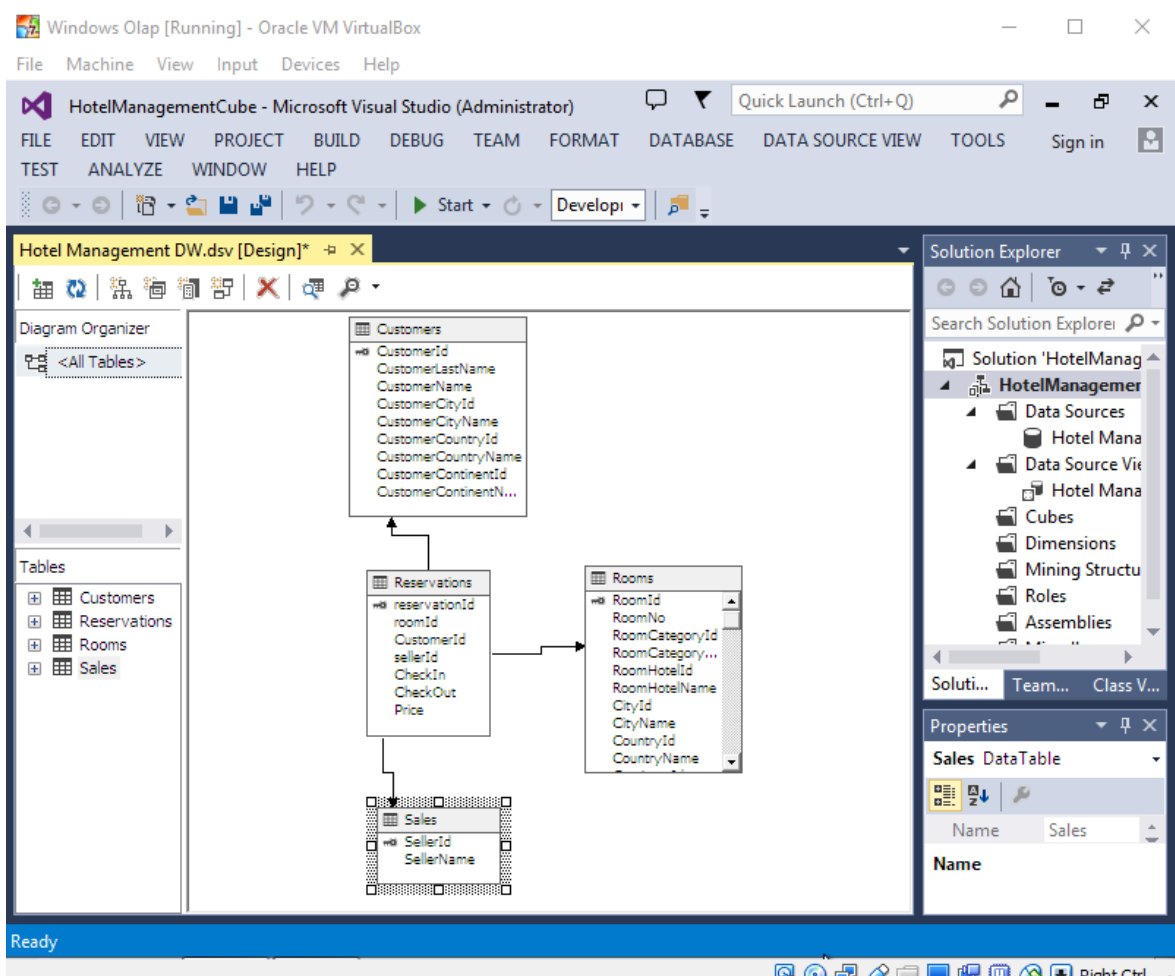
Σε αυτή την οθόνη όπως φαίνεται στην εικόνα 50 μπορούμε να δώσουμε όνομα στο Data Source View και από κάτω στο preview να δούμε την τελική του μορφή με τους πίνακες που ενσωματώνει. Πατάμε finish και ολοκληρώνουμε την δημιουργία του Data Source View.



Εικόνα 50 "Πίνακες Data source View"

Πατώντας open στο data source view μπορούμε να δούμε τους πίνακες και τις συσχετίσεις τους.

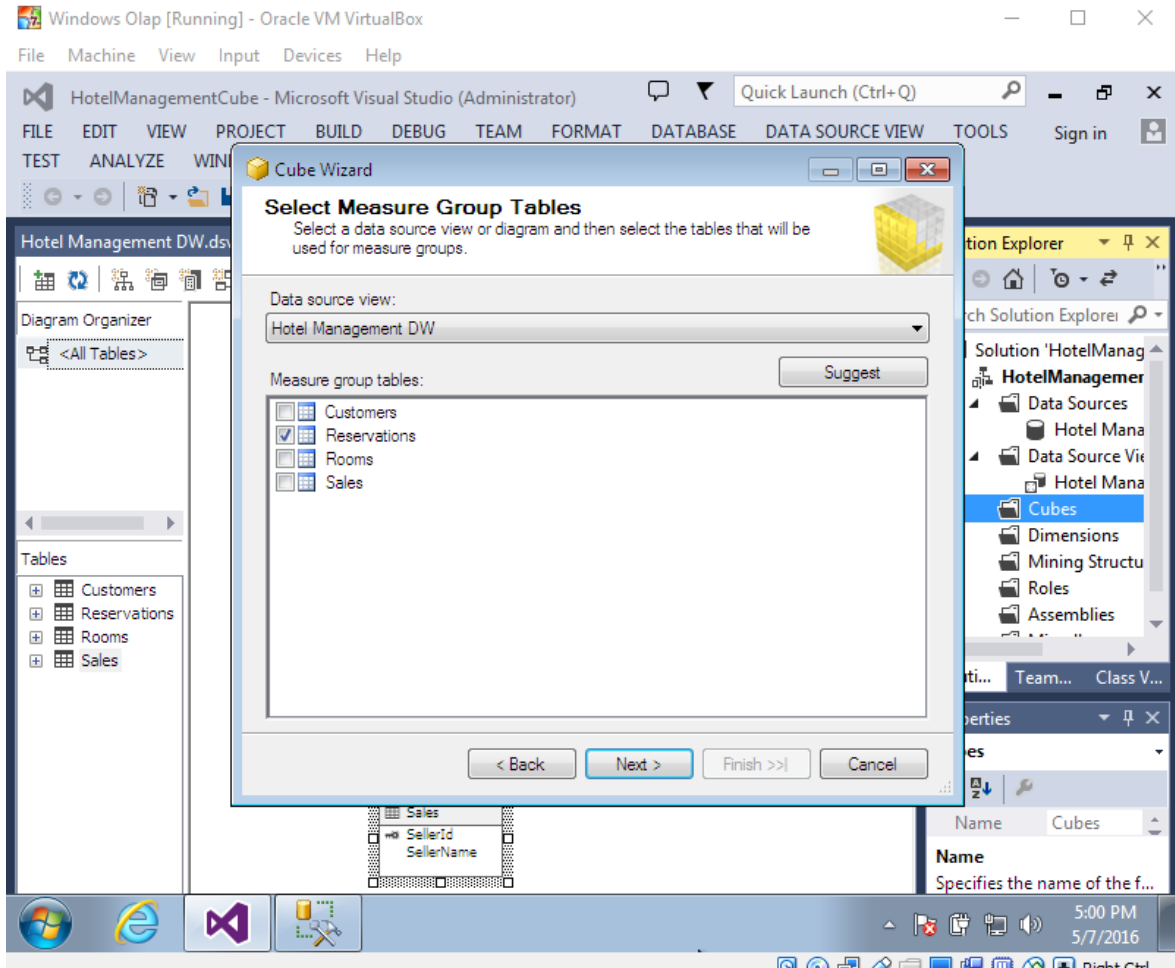
Στο preview στην εικόνα 50 φαίνονται οι πίνακες Reservations, Customers, Rooms και Sales που θα δημιουργηθούν στο View. Όπως παρατηρούμε είναι ένας πίνακας γεγονότων με τρεις πίνακες διαστάσεων το οποίο στην επόμενη εικόνα θα το δούμε σε μορφή σχήματος αστέρα.



Εικόνα 51 "Data source View Browse"

Στην εικόνα 51 φαίνεται αναλυτικά το σχήμα αστέρα στο design του data source view που μόλις δημιουργήσαμε. Ο πίνακας reservations που αφορά την κράτηση ενός δωματίου θα είναι ο πίνακας fact και γύρο του οι διαστάσεις Sales, Rooms και Customers. Αντίστοιχα είναι οι πωλητές, τα δωμάτια και οι πελάτες το καθένα συνδέεται με foreign key με τον πίνακα κρατήσεων. Πιο κάτω θα προσθέσουμε μία ακόμη διάσταση την Date που θα περιέχει την ημερομηνία που έγινε η κράτηση σε μορφή ιεραρχημένη.

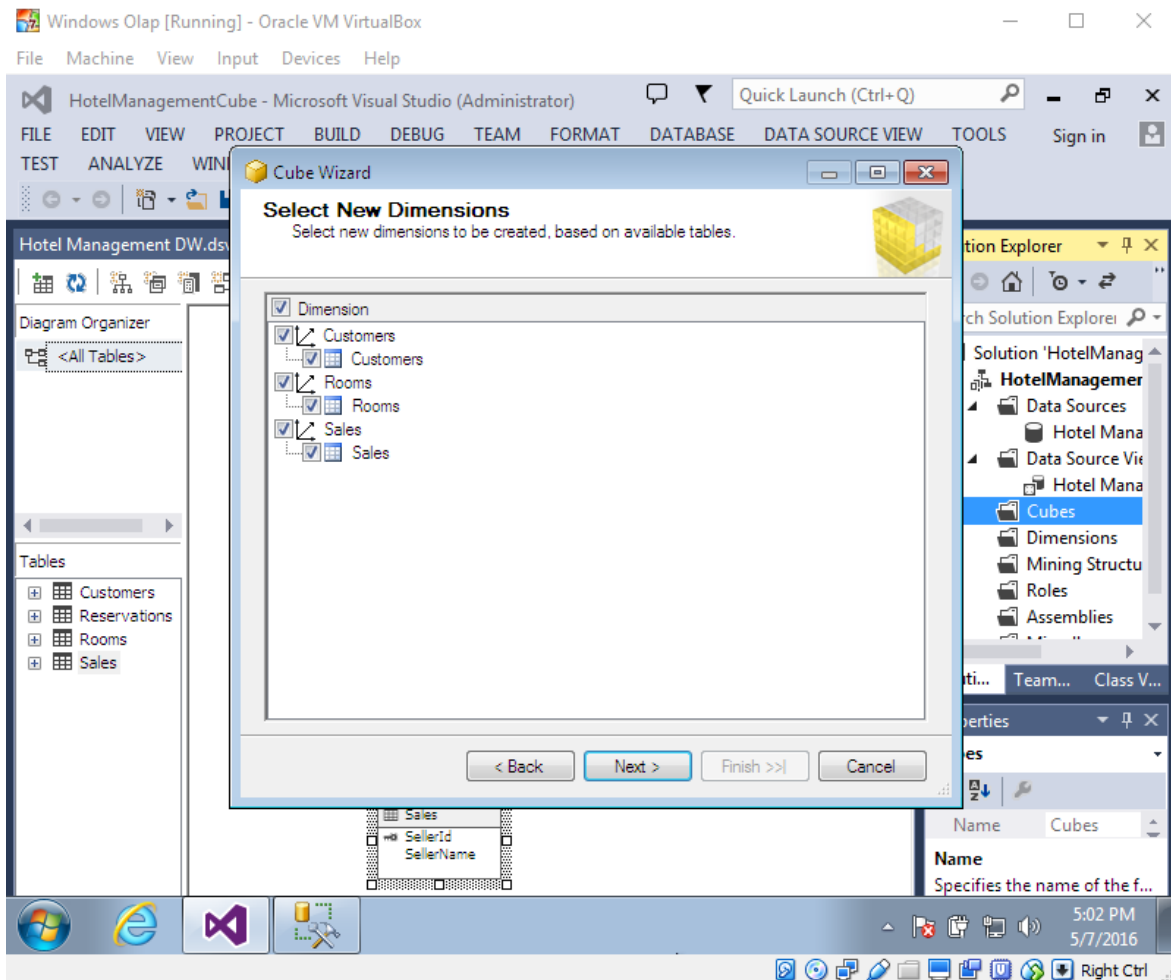
Το τρίτο βήμα είναι να δημιουργήσουμε τον κύβο. Πατώντας new cube στο πεδίο cube και επιλέγοντας use existing tables μας οδηγεί στην οθόνη για να επιλέξουμε τα measures του κύβου, το πίνακα γεγονότων. Χρησιμοποιώντας τον wizard προχωρούμε στην δημιουργία του κύβου step by step.



Εικόνα 52 "OLAP Cube Measure"

Σε αυτό το σημείο επιλέγουμε το data source view που επιθυμούμε να χρησιμοποιήσουμε και τον πίνακα reservations που θα είναι ο λεγόμενος fact table όπως βλέπουμε στην εικόνα 52 ορίζοντας με αυτόν τον τρόπο τις μετρικές του κύβου. Το κουμπί Suggest βάση κάποιων συγκεκριμένων κριτηρίων προτείνει ποιόν πίνακα μπορούμε να επιλέξουμε για πίνακα γεγονότων.

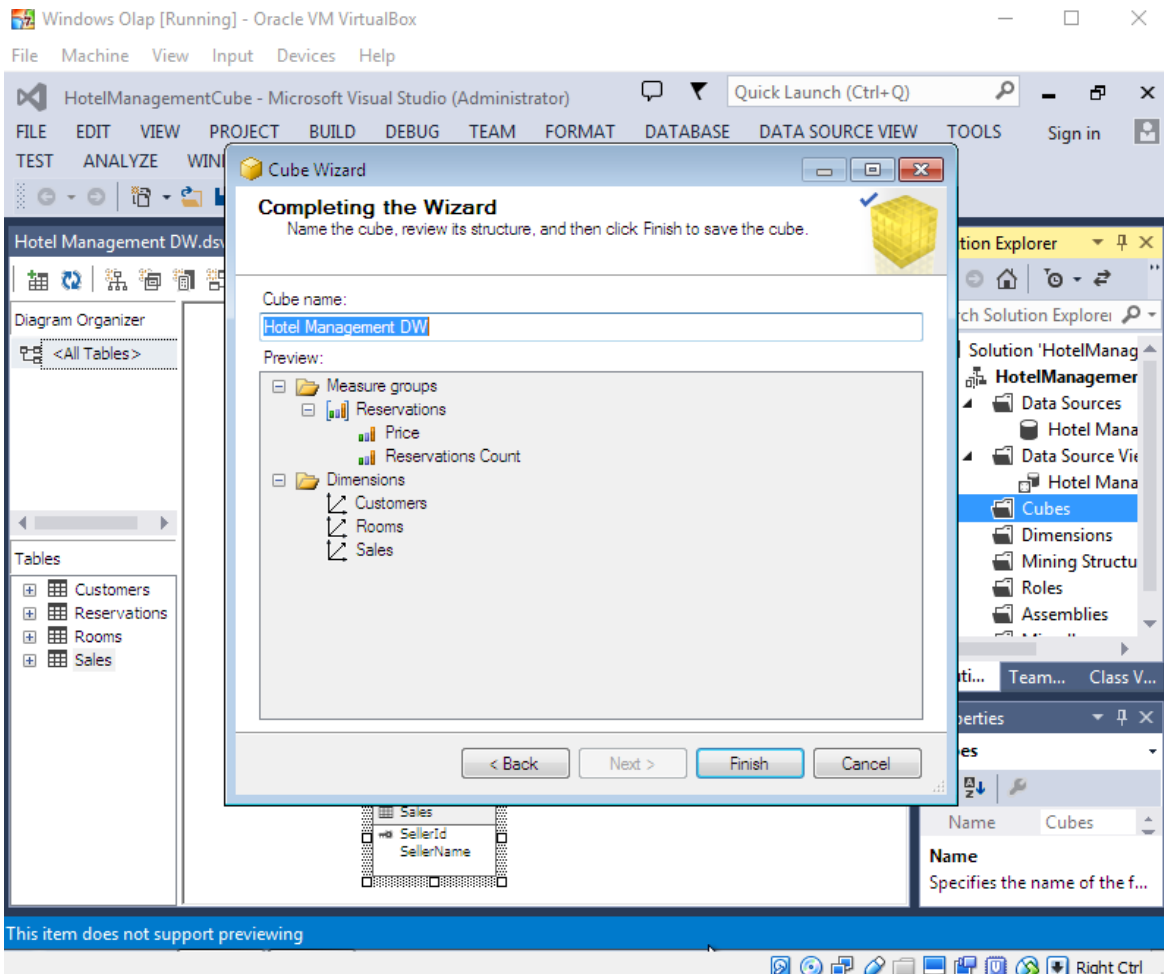
Στο επόμενο βήμα θα προσθέσουμε τις διαστάσεις του κύβου. Πατώντας next πηγαίνουμε στην επόμενη οθόνη όπως βλέπουμε στην εικόνα 53.



Εικόνα 53 "OLAP Cube Dimensions"

Το visual studio αναγνωρίζει και φορτώνει τους πίνακες διαστάσεων Customers, Rooms και Sales. Μπορούμε να κάνουμε check σε όποιες διαστάσεις θέλουμε να συμπεριλάβουμε, στο παράδειγμα μας κρατάμε και τις τρεις διαστάσεις.

Πατάμε next και πηγαίνουμε στην τελευταία οθόνη όπου το Visual Studio μας αναπαριστά τις μετρικές και τις διαστάσεις που θα δημιουργηθούν.

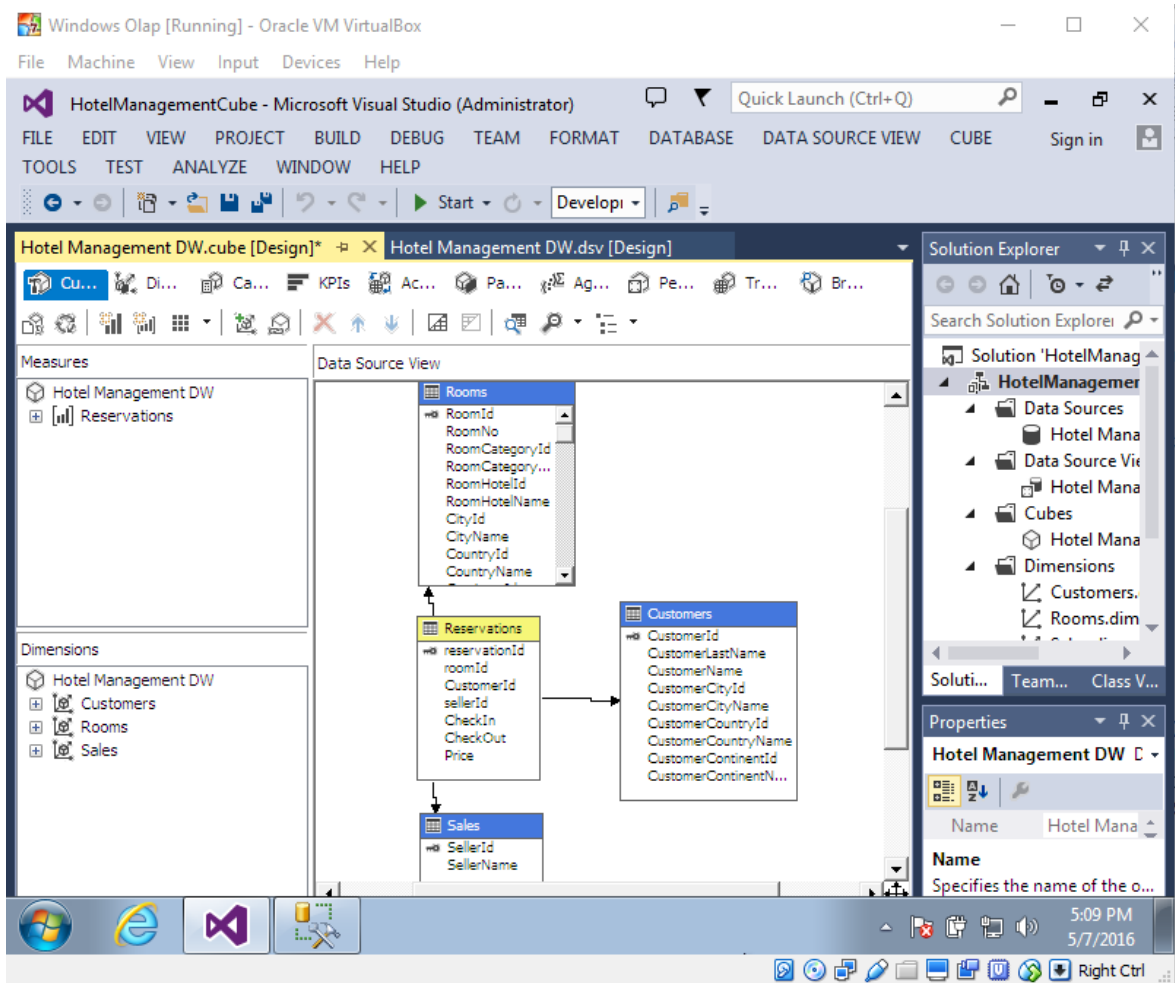


Εικόνα 54 "OLAP Cube Measures and Dimensions"

Στο Cube Name ορίζουμε το όνομα του κύβου και στο preview παρατηρούμε τα measures και τα dimensions. Τέλος όπως φαίνεται στην εικόνα 54 πατάμε finish για να δημιουργηθεί ο κύβος.

Πατώντας finish μας ανοίγει το Design για να δούμε το σχήμα του κύβου με τον πίνακα γεγονότων τις διαστάσεις και τις συσχετίσεις τους όπως βλέπουμε στην εικόνα 55.

Με κίτρινο χρώμα αναπαριστάται ο πίνακας γεγονότων και με το μπλε χρώμα οι πίνακες διαστάσεων.

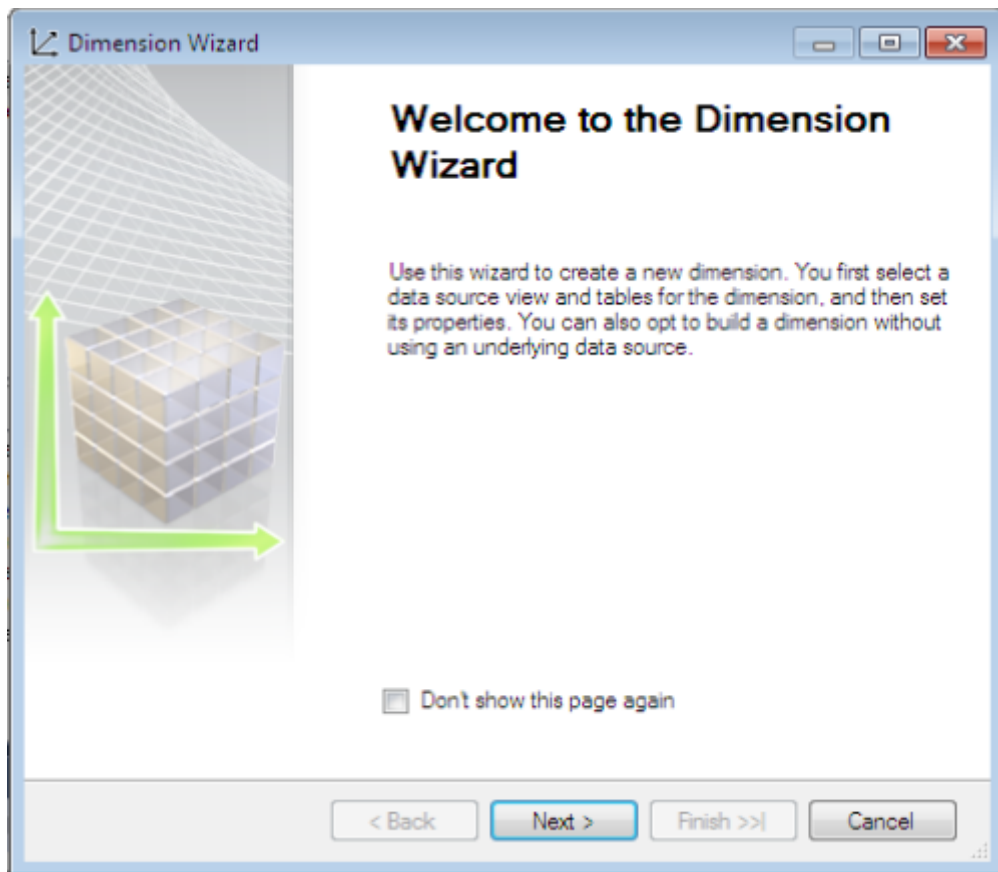


Εικόνα 55 "OLAP Cube Design Star Schema"

Σε αυτό το σημείο πρέπει να παρατηρήσουμε ότι ίσως μια από τις πιο σημαντικές διαστάσεις του OLAP η διάσταση Time δεν υπάρχει. Θα πρέπει να την δημιουργήσουμε από την αρχή. Η διάσταση Time θα είναι ένας πίνακας με κύριο κλειδί την ημερομηνία και τα επόμενα κελιά θα στεγάζουν πληροφορία που θα κατηγοριοποιεί την ημερομηνία.

Θα δημιουργήσουμε την διάσταση Time πάνω στο κελί check in του πίνακα reservations. Με αυτό τον τρόπο θα μπορούμε να εξάγουμε πληροφορία για το πότε γίνεται μια κράτηση.

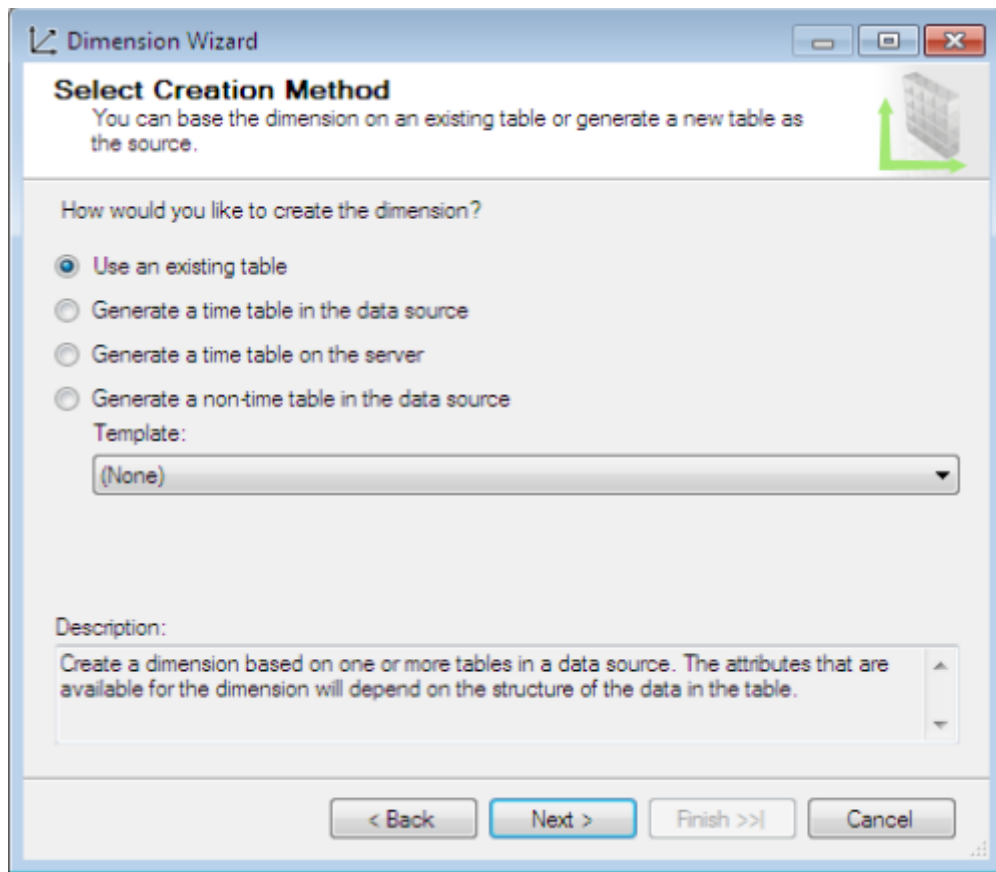
Στο menu dimensions επιλέγουμε new Dimension. Το Visual Studio θα μας ανοίξει τον wizard δημιουργίας καινούριας διάστασης όπως βλέπουμε στην εικόνα 56.



Εικόνα 56 "Dimension Create Wizard"

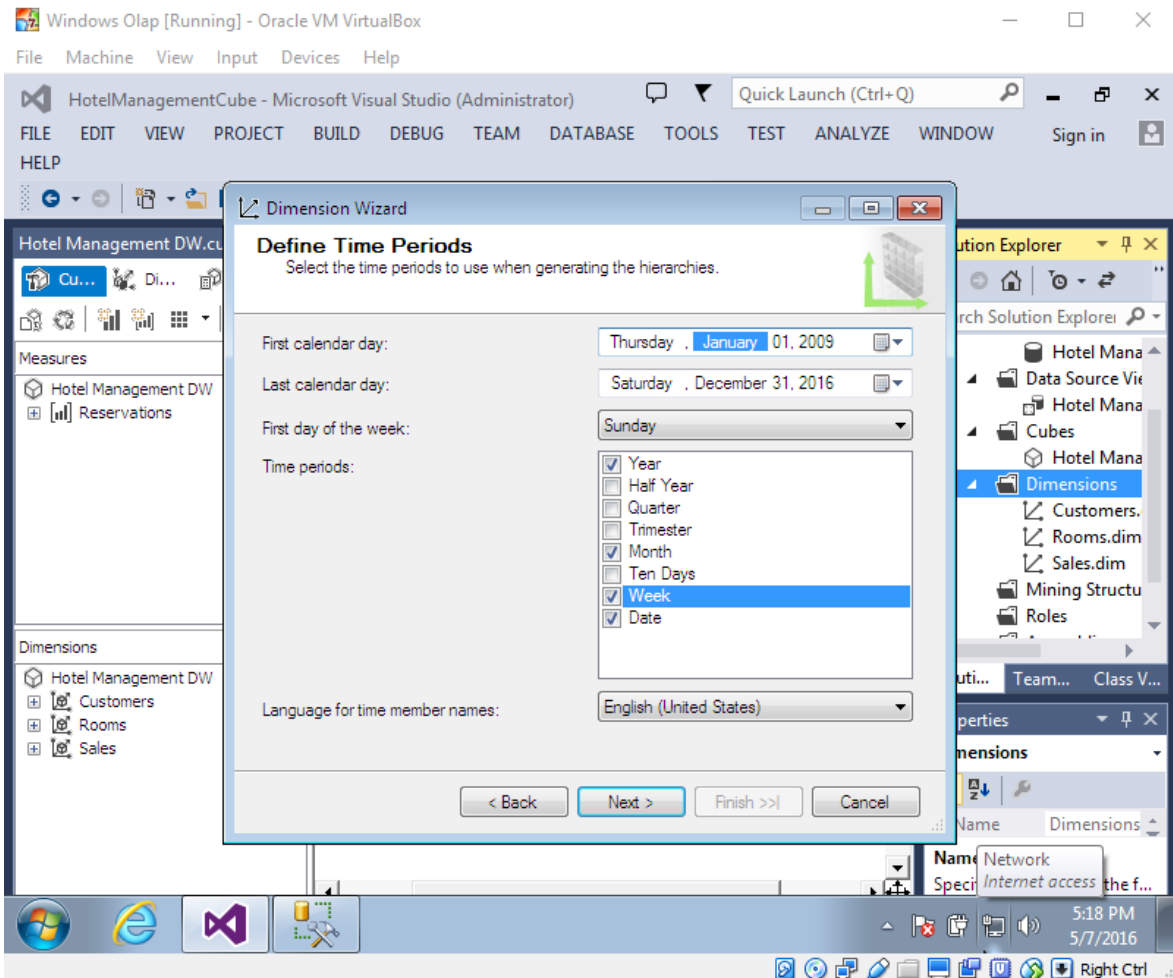
Πατάμε next. Στην επόμενη οθόνη της εικόνας 57 ο wizard μας παρέχει δυνατότητες όπως το να χρησιμοποιήσουμε έναν ήδη υπάρχον πίνακα ή να δημιουργήσουμε ένα time table στο Data source ή να δημιουργήσουμε ένα non time table. Στα templates υπάρχουν κάποια προτεινόμενα είδη διαστάσεων που μπορεί να βολεύουν να χρησιμοποιήσουμε. Στο παράδειγμα μας χρειαζόμαστε ένα time table on the server το επιλέγουμε και πατάμε next.





Εικόνα 57 "Select Creation Method"

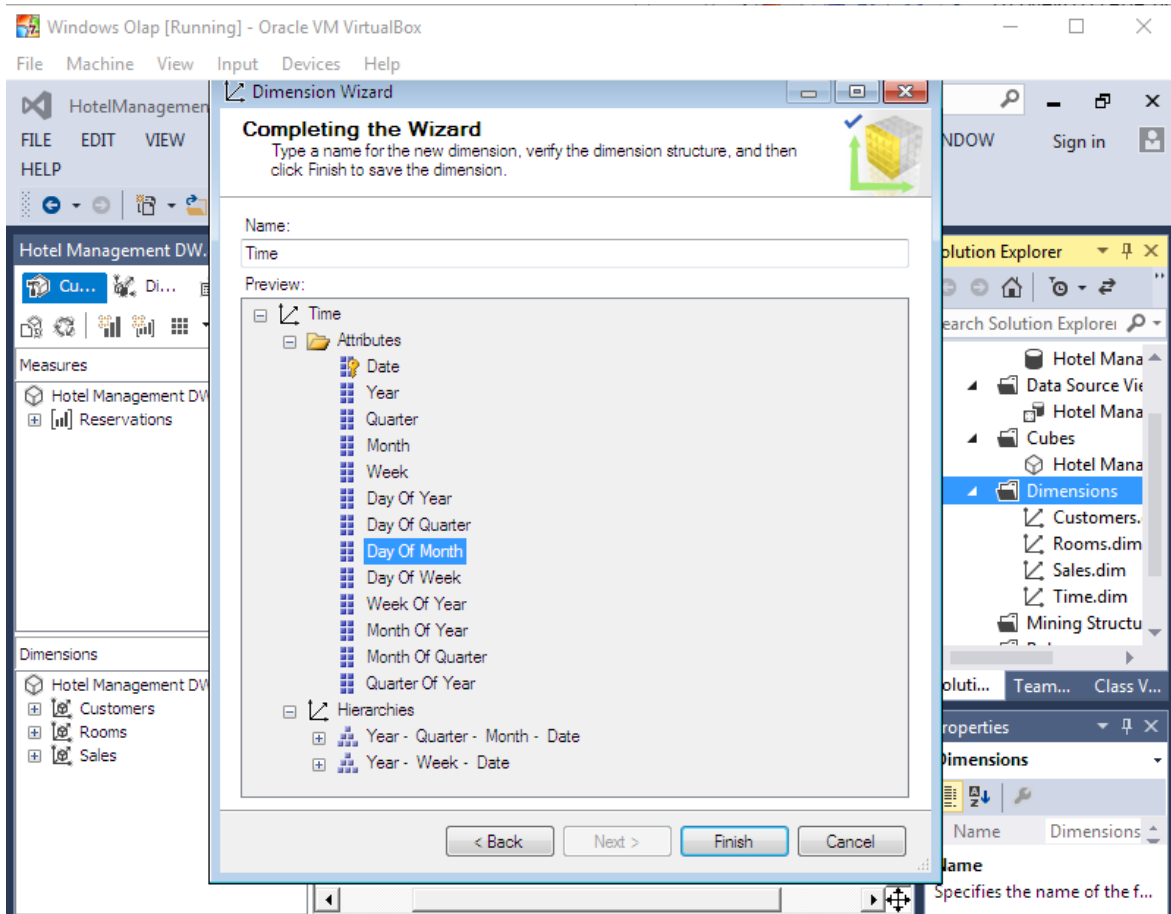
Σε αυτό το βήμα επιλέγουμε το βάθος της ιεράρχησης που επιθυμούμε στην διάσταση μας, Year , Quarter, Month, Week, Date. Στην εικόνα 58 κάνουμε check στα time periods που μας ενδιαφέρουν να δημιουργήσει ο wizard. Στο language επιλέγουμε την μορφή που θα έχει η ημερομηνία στον πίνακα ανάλογα με την χώρα που μας ενδιαφέρει. Στην Ελλάδα χρησιμοποιούμε την μορφή dd/mm/yyyy ενώ στην Αμερική χρησιμοποιούν την μορφή yyyy/mm/dd. Στο παράδειγμα μας χρησιμοποιούμε το Αμερικάνικο στυλ γραφής.



Εικόνα 58 "Time Dimension"

Είναι σημαντικό το εύρος των Ημερομηνιών που θα δώσουμε να περιλαμβάνει και την πρώτη και την τελευταία ημερομηνία που έγινε η πρώτη και τελευταία κράτηση που έχουμε στην αποθήκη δεδομένων. Αυτό το ορίζουμε στο First και Last calendar day.

Στο τέλος μας εμφανίζει στο preview την διάσταση που θα δημιουργηθεί με τις ιδιότητες τις και τις ιεραρχίες όπως φαίνεται στην εικόνα 59.

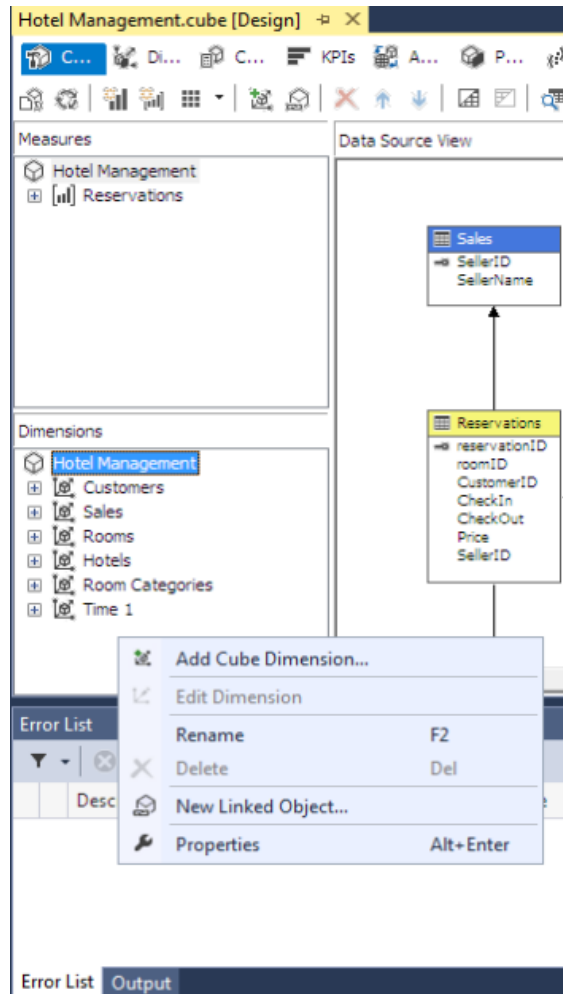


Εικόνα 59 "Time Dimension Attributes and Hierarchies"

Βλέπουμε στην εικόνα 59 ότι το κύριο κλειδί της διάστασης θα είναι το date. Τα υπόλοιπα πεδία κατηγοριοποιούν το πεδίο date ως προς την θέση του μέσα στην εβδομάδα, το μήνα, το χρόνο, το εξάμηνο. Το Wizard δημιουργεί αυτόματα τις ιεραρχίες της διάστασης Time.

Το τελευταίο πράγμα που πρέπει να κάνουμε είναι να συνδέσουμε την διάσταση με τον πίνακα γεγονότων και κατ' επέκταση το date του Time με το CheckIn του Reservations.

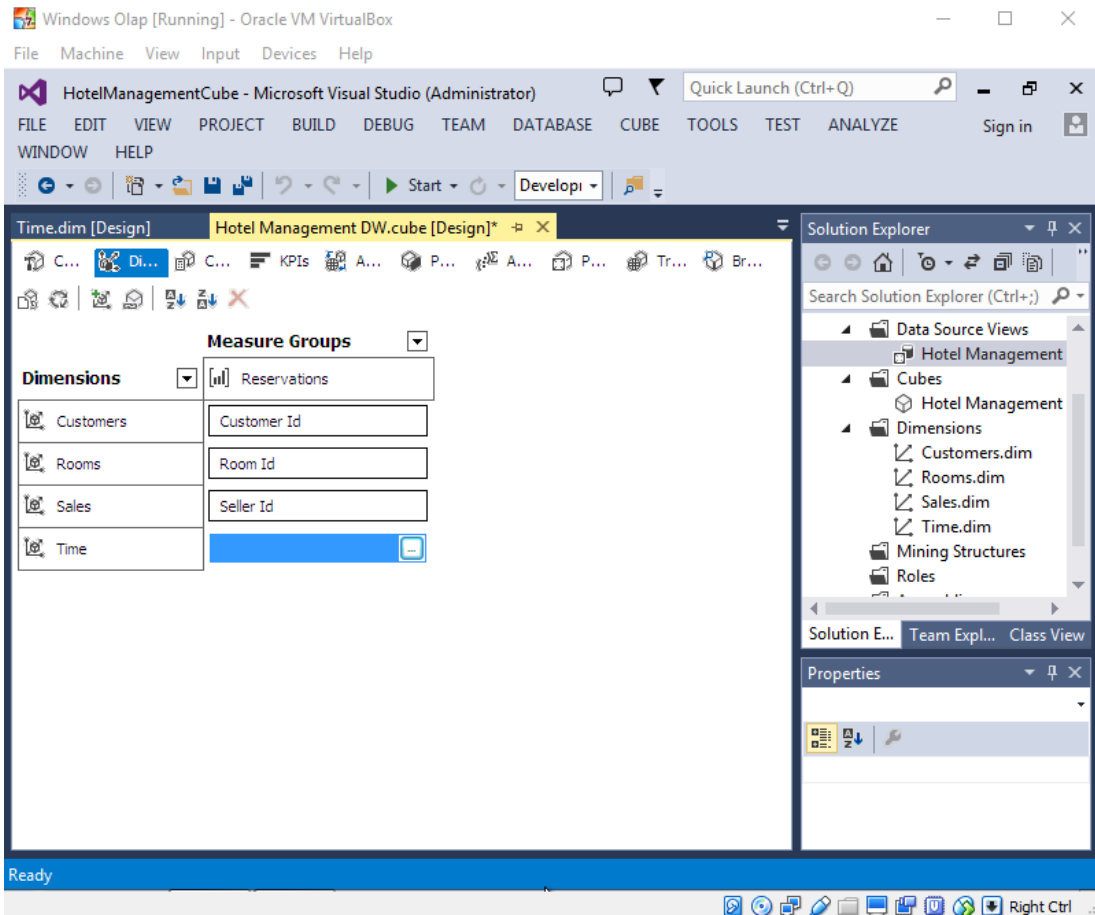
Στο Design του κύβου πηγαίνουμε στις διαστάσεις και επιλέγουμε add Cube Dimension και επιλέγουμε την διάσταση Time όπως βλέπουμε στην εικόνα 60.



Εικόνα 60 "Cube Design - Dimensions – Add Cube Dimension"

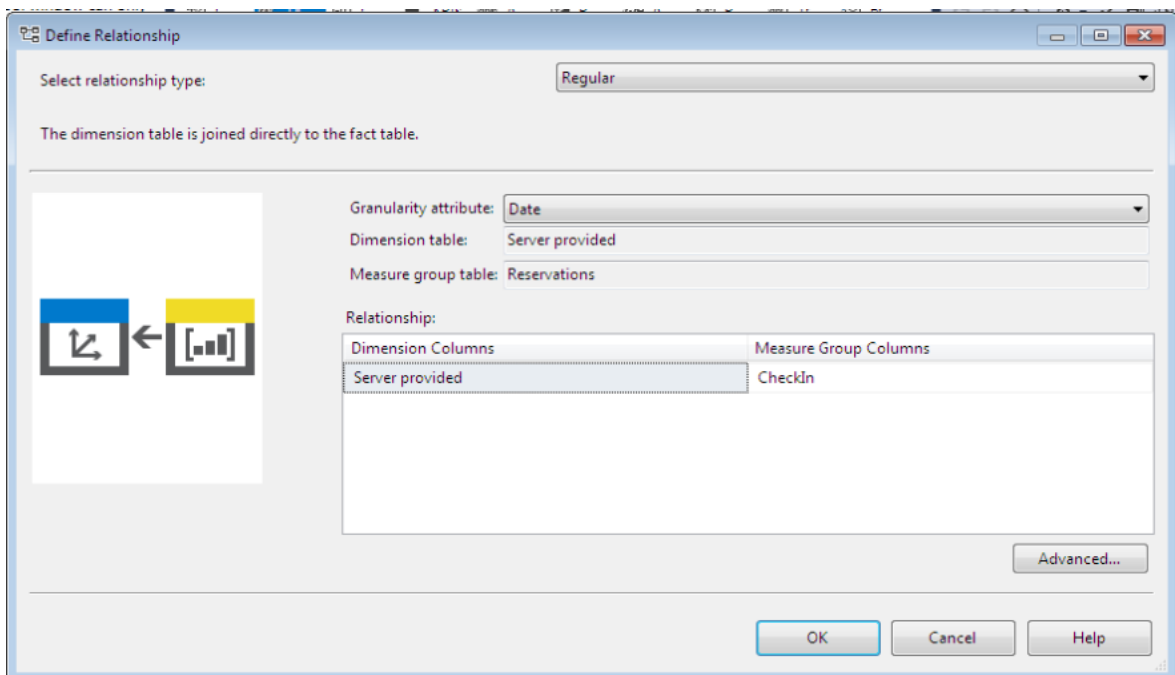
Αφού εισάγουμε την διάσταση πηγαίνουμε στο menu dimensions μέσα στο design του κύβου για να συσχετίσουμε την διάσταση με το πίνακα γεγονότων. Πρέπει το date πεδίο της διάστασης time να συνδεθεί με foreign key με το Check In πεδίο του πίνακα Reservations. Στην εικόνα 61 κάτω από τα measure groups επιλέγουμε το κουμπί που βρίσκεται δίπλα από την γραμμή time.

## Πτυχιακή εργασία του φοιτητή Δουλάμη Δημήτρη



Εικόνα 61 "Time Dimension - Measure"

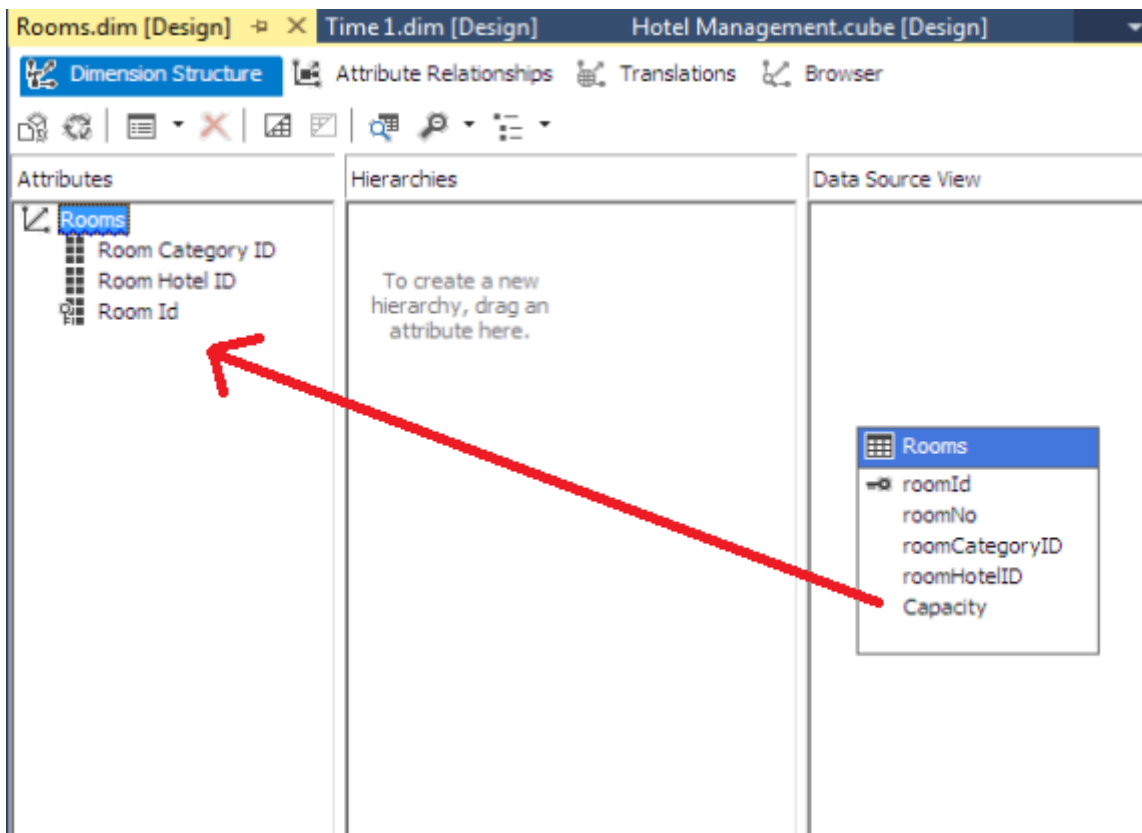
Πατώντας το κουμπί με τις τρεις τελείες που είναι επιλεγμένο με μπλε χρώμα θα ανοίξει το wizard που βλέπουμε στην επόμενη εικόνα (εικόνα 62).



Εικόνα 62 "Wizard σύνδεσης Dimension - Measure"

Στο relationship type επιλέγουμε Regular. Στο relationship συνδέουμε το πεδίο Time Date με το πεδίο CheckIn και η διάσταση Time είναι έτοιμη. Στο τέλος πατάμε πάνω στην διάσταση Time και επιλέγουμε να γίνει process. Σε αυτό το σημείο έχουμε ολοκληρώσει την δημιουργία της διάστασης time.

Το επόμενο που πρέπει να κάνουμε είναι να φορτώσουμε κάποια attribute στην διάσταση rooms. Στο design της διάστασης rooms και συγκεκριμένα στο dimension structure μπορούμε να παρατηρήσουμε ότι σαν attribute φόρτωσε μόνο το roomId ενώ στο data source view έχει όλα τα fields.

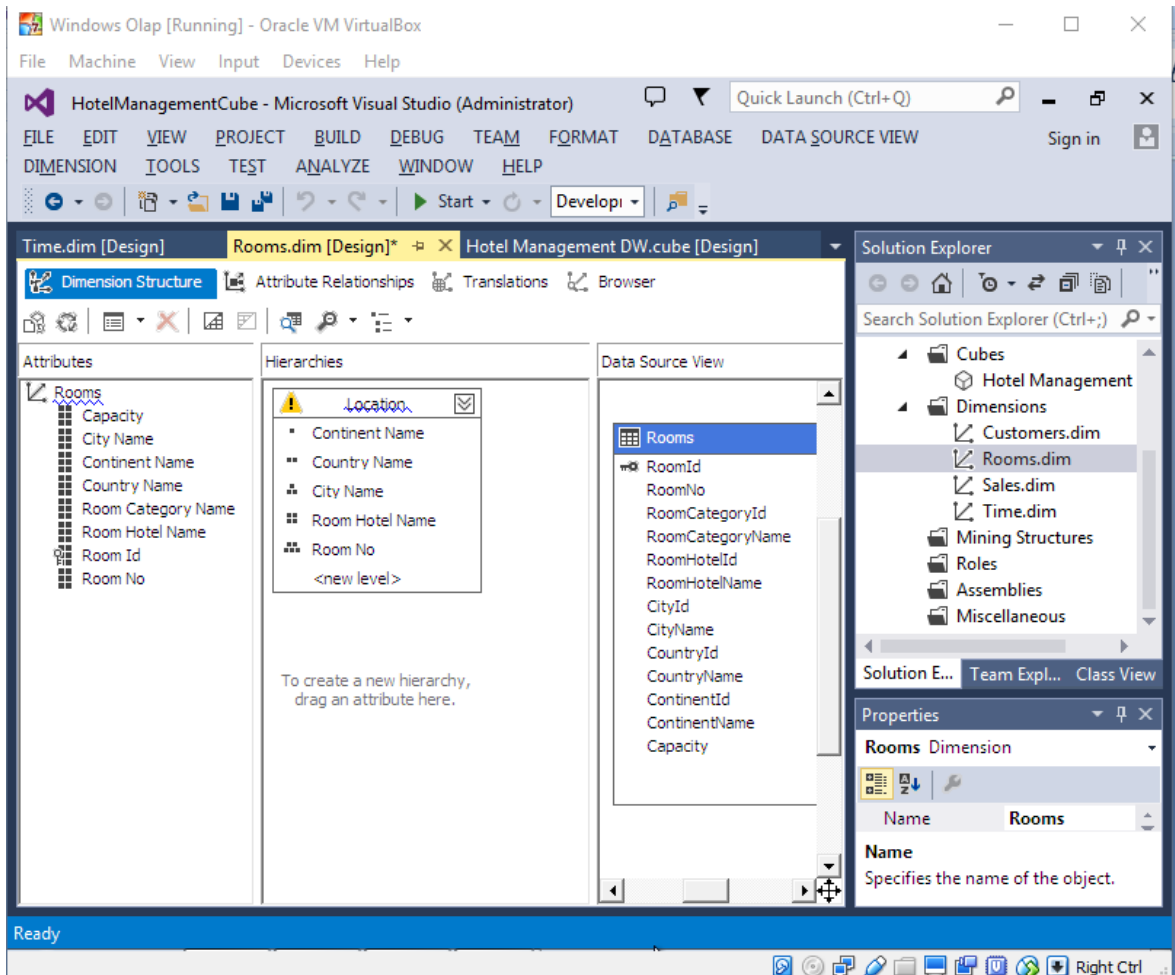


Εικόνα 63 "Add attribute to dimension"

Μπορούμε να πάρουμε τα πεδία που έχει στο data source view και επιθυμούμε να βάλουμε σαν attributes και να τα κάνουμε drag and drop μέσα στα attributes όπως βλέπουμε στην εικόνα 63. Στο παράδειγμα μας κάνουμε drag and drop στο attributes το capacity από το data source View, που είναι η χωρητικότητα του κάθε δωματίου, ώστε να το ενσωματώσουμε στην διάσταση και κατ' επέκταση στον κύβο.

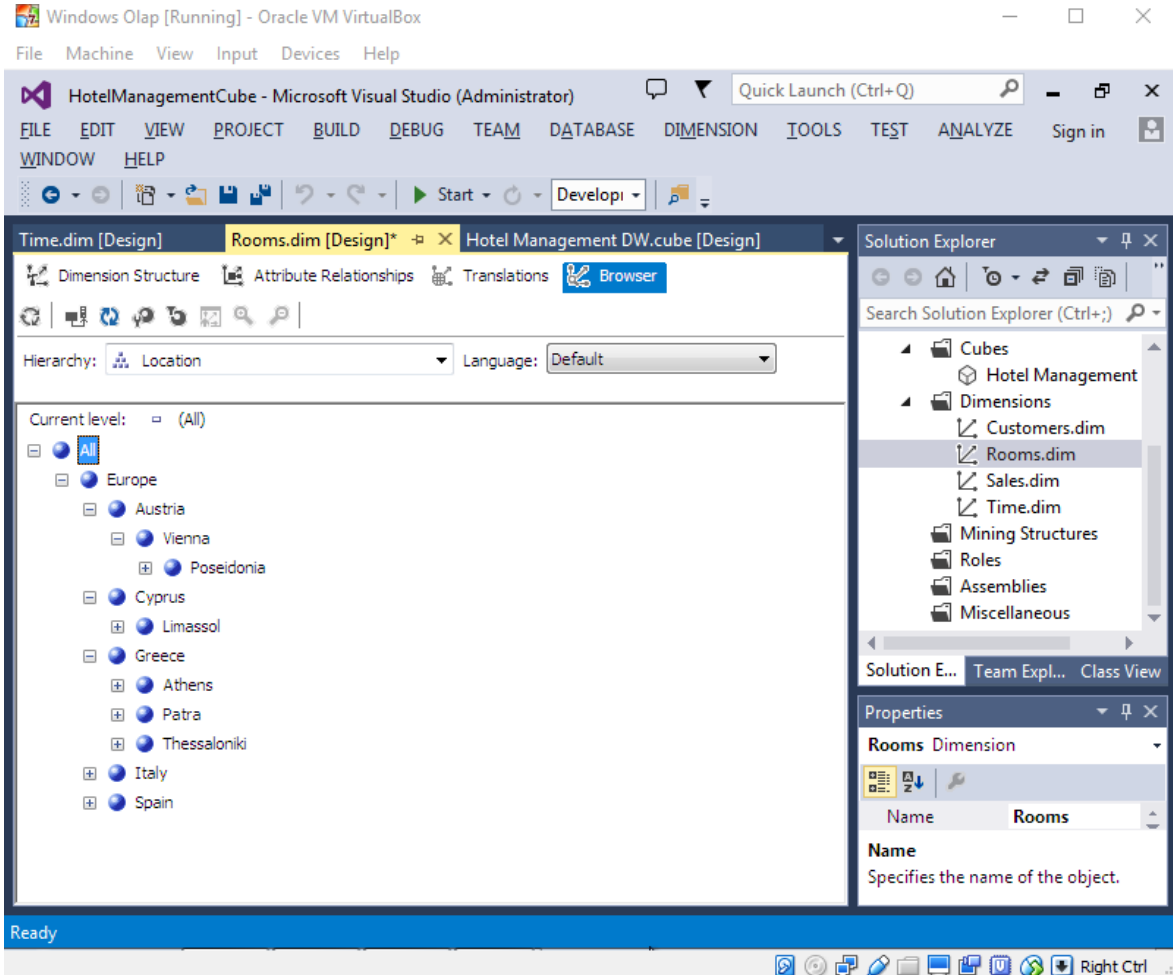
Το επόμενο βήμα είναι να δημιουργήσουμε μια ιεραρχία ανάμεσα στο room, hotel, city, country και continent για να μπορούμε να έχουμε μια καλή οπτικοποίηση και να μπορούμε να κάνουμε drill-down ή roll-up κατά βούληση. Από το πεδίο attribute

κάνουμε drag and drop στο πεδίο hierarchies και δημιουργούμε την ιεραρχία όπως βλέπουμε στην εικόνα 64.



Εικόνα 64 "Rooms Dimension Δημιουργία Ιεραρχίας"

Στο τέλος αποθηκεύουμε και κάνουμε process και run την διάσταση. Στο browser στην εικόνα 65 μπορούμε να δούμε πως δημιουργήθηκε η ιεραρχία σύμφωνα με τα πεδία που ορίσαμε. Το Design του Visual Studio κάνει group τα δεδομένα σε ιεραρχίες που μπορούν να γίνουν expand και collapse κάνοντας στην ουσία drill through ή roll out στην διάσταση.

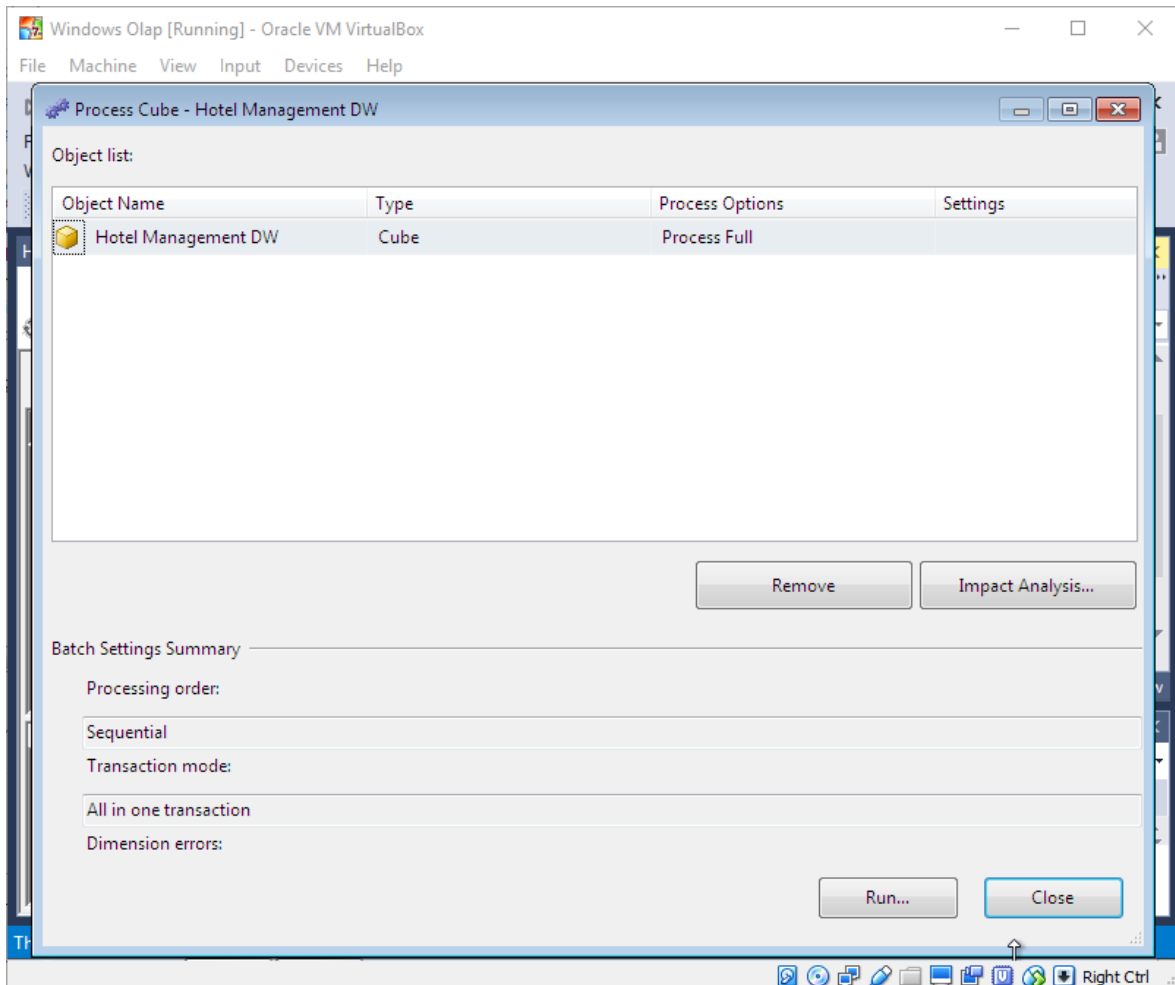


Εικόνα 65 "Rooms Dimension Browser"

Την ίδια τεχνική πρέπει να ακολουθήσουμε και για την διάσταση customers που έχει και αυτή αποθηκευμένα στοιχεία προορισμού των επισκεπτών για την ιεράρχηση των ιδιοτήτων.

Στο τέλος είναι σημαντικό να μην παραλείψουμε την διαδικασία process και run καθώς μόνο τότε θα δομηθεί και θα δημιουργηθεί ο κύβος και οι διαστάσεις. Κάθε φορά που πατάμε process και run τον κύβο αυτός επεξεργάζεται και γίνεται deploy στον Analysis Server.





Εικόνα 66 "OLAP Cube Process"

Αφού έχουμε ετοιμάσει τον κύβο και τις διαστάσεις είναι η ώρα να κάνουμε process τον κύβο. Επιλέγουμε τον κύβο – επιλέγουμε process και στο παράθυρο που ανοίγει επιλέγουμε run. Ο κύβος σε αυτό το σημείο είναι έτοιμος για να γίνει η ανάλυση των δεδομένων από τους αναλυτές.

Στο menu Browse του design μπορούμε να δούμε τον κύβο και να δημιουργήσουμε reports πάνω στα δεδομένα των διαστάσεων και των μετρικών που επιθυμούμε.

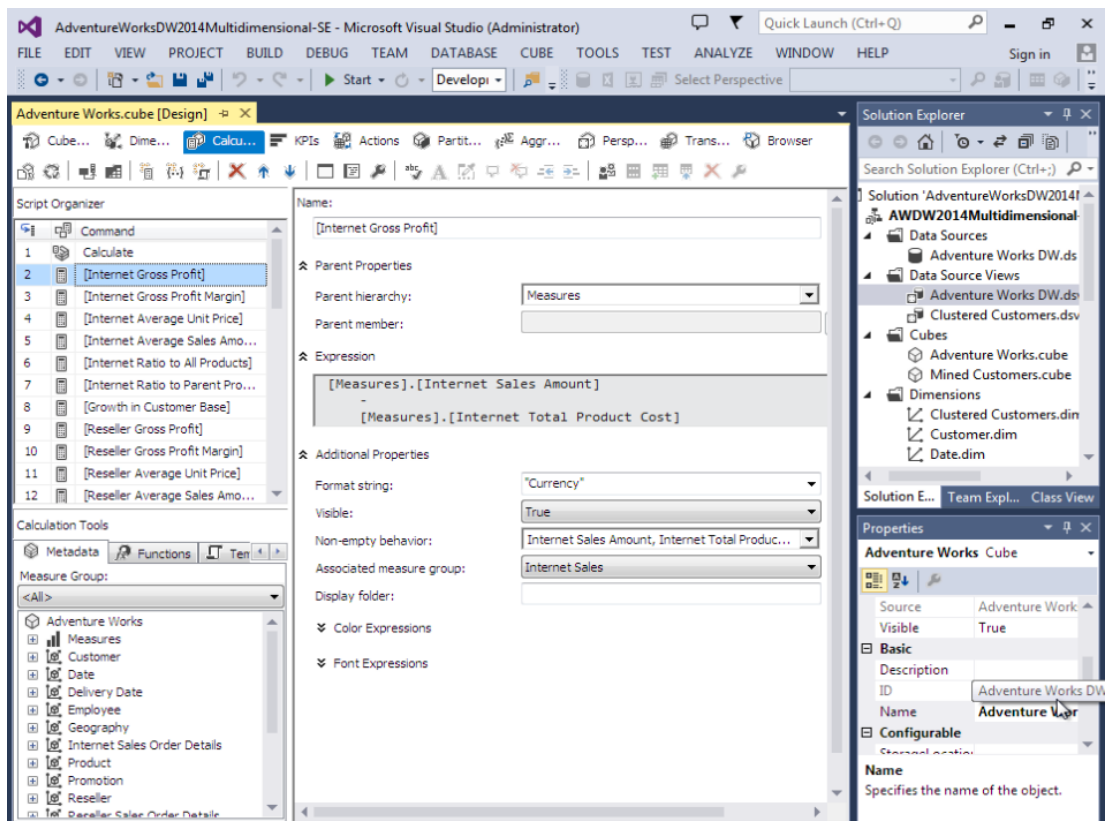
Ο κύβος μόλις γίνει processed – build και deployed μπορεί να φορτωθεί και στο SQL Management Studio για να αναλυθεί ή και να γραφούν MDX queries. Στο Management Studio κάνουμε ένα νέο connect σε Server Type : Analysis Services και φορτώνει στον Object Explorer τον κύβο Hotel Management Cube. Πατώντας New Query μας δίνει τις επιλογές να γράψουμε DMX, MDX και XMLA scripts.

### 3.7 SSAS OLAP Cube Adventure Works

Σε αυτό το κεφάλαιο θα αναλύσουμε τον κύβο Adventure Works στο Visual Studio. Ο κύβος Adventure Works θα πάρει τα δεδομένα του από την αποθήκη δεδομένων Adventure Works DW 2014 που έχουμε αποθηκευμένη στον SQL Server. Το Analysis Services που θα χρησιμοποιήσουμε για να μελετήσουμε την τεχνολογία OLAP είναι το Adventure Works DW Multidimensional 2014 το οποίο το προσφέρει η Microsoft. Το project το έχουμε κατεβάσει από την Microsoft στο εικονικό μηχάνημα στο Virtual Box - στο desktop - στο φάκελο AW multidimensional objects SQL Server 2014.

#### 3.7.1 SSAS Calculations

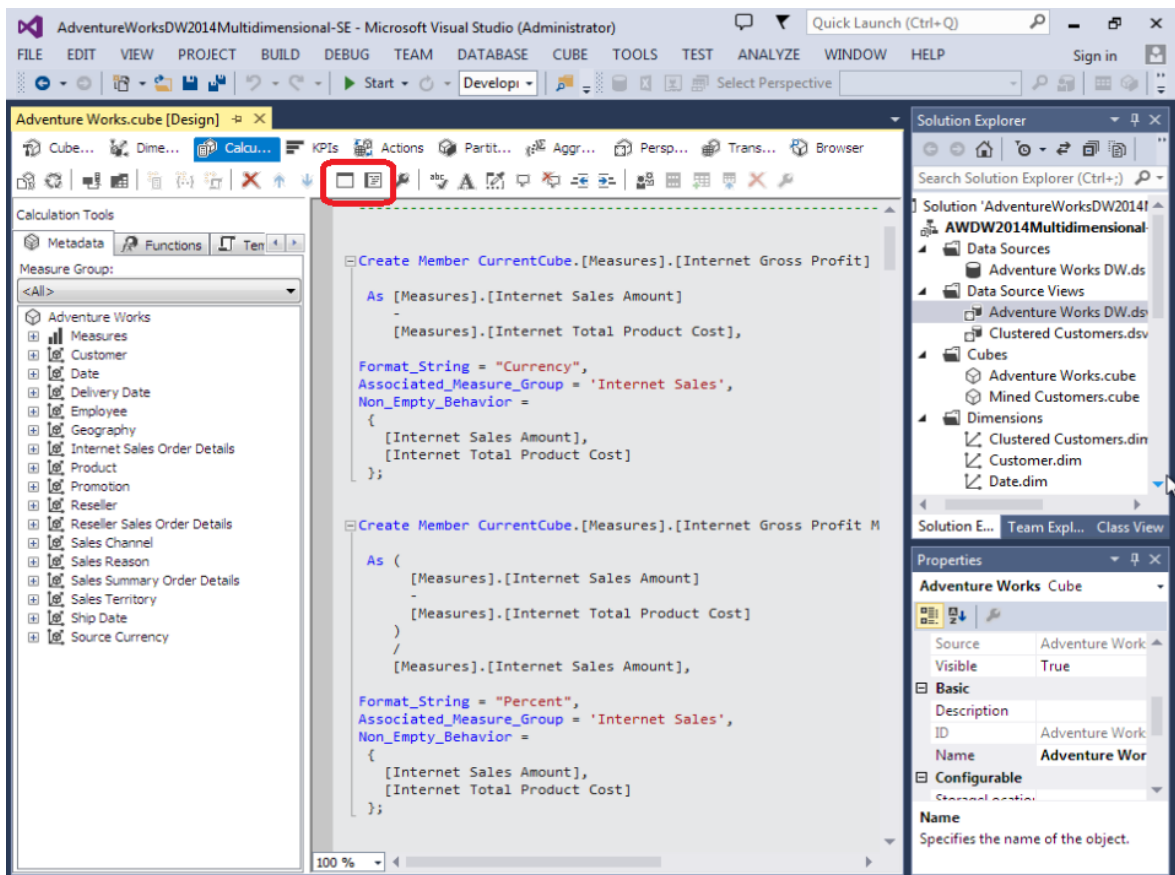
Το πρώτο πράγμα που θα αναλύσουμε είναι το μενού calculations το τρίτο tab στο Design του κύβου. Σε αυτό το μενού έχουμε την δυνατότητα να δημιουργήσουμε calculated members. Τα calculated members είναι δημιουργημένα πεδία μέσα από τα δεδομένα του κύβου τα οποία πληρούν κάποιες προϋποθέσεις και εφαρμόζουν κάποιους κανόνες που έχουμε ζητήσει. Στην εικόνα 67 βλέπουμε το calculated member που έχουμε δημιουργήσει Internet Gross Profit.



Εικόνα 67 "Calculations – Calculated Members"

Στο πάνω μέρος βλέπουμε το όνομα του calculated member. Στο parent hierarchy βλέπουμε ότι ανήκει στην ιεραρχία measures και αποτελεί μια μετρική του κύβου. Το σημαντικό σημείο είναι στο Panel Expression, σε αυτό το σημείο ορίζουμε τον κανόνα που θα δημιουργήσει το calculated member. Στο παράδειγμα μας το Internet gross profit δημιουργείται αν από την τιμή του πεδίου Internet Sales Amount αφαιρέσουμε το πεδίο Internet Total product Cost. Δημιουργήσαμε δηλαδή ένα πεδίο στο κύβο που κρατάει το μικτό κέρδος από τις πωλήσεις από το Internet και αυτό το βρίσκουμε αν αφαιρέσουμε από το πεδίο που κρατάει το ποσό της πώλησης το πεδίο που έχει το κόστος του προϊόντος. Τέλος το format του member ορίζεται ως currency.

Τα calculated member προσδίδουν επιχειρηματική ευφυΐα στον κύβο μας σύμφωνα με τις ανάγκες για παραπάνω πληροφορία.



Εικόνα 68 "Calculations – Form View – Script View"

Κάτι σημαντικό που πρέπει να αναφέρουμε σε αυτό το σημείο είναι τα δύο κουμπιά που υπάρχουν μέσα στο calculations τα οποία κάνουν toggle μεταξύ Form View και Script View των calculated members. Το Form View φαίνεται στην προηγούμενη εικόνα (εικόνα 67). Το Script View της εικόνας 68 μας δείχνει τον κώδικα MDX που χρειάζεται για να δημιουργηθεί ένα calculated member. Ανάλογα την πολυπλοκότητα του calculated member μπορεί να προτιμήσουμε την δημιουργία του μέσω κώδικα MDX.

### 3.7.2 SSAS KPIs

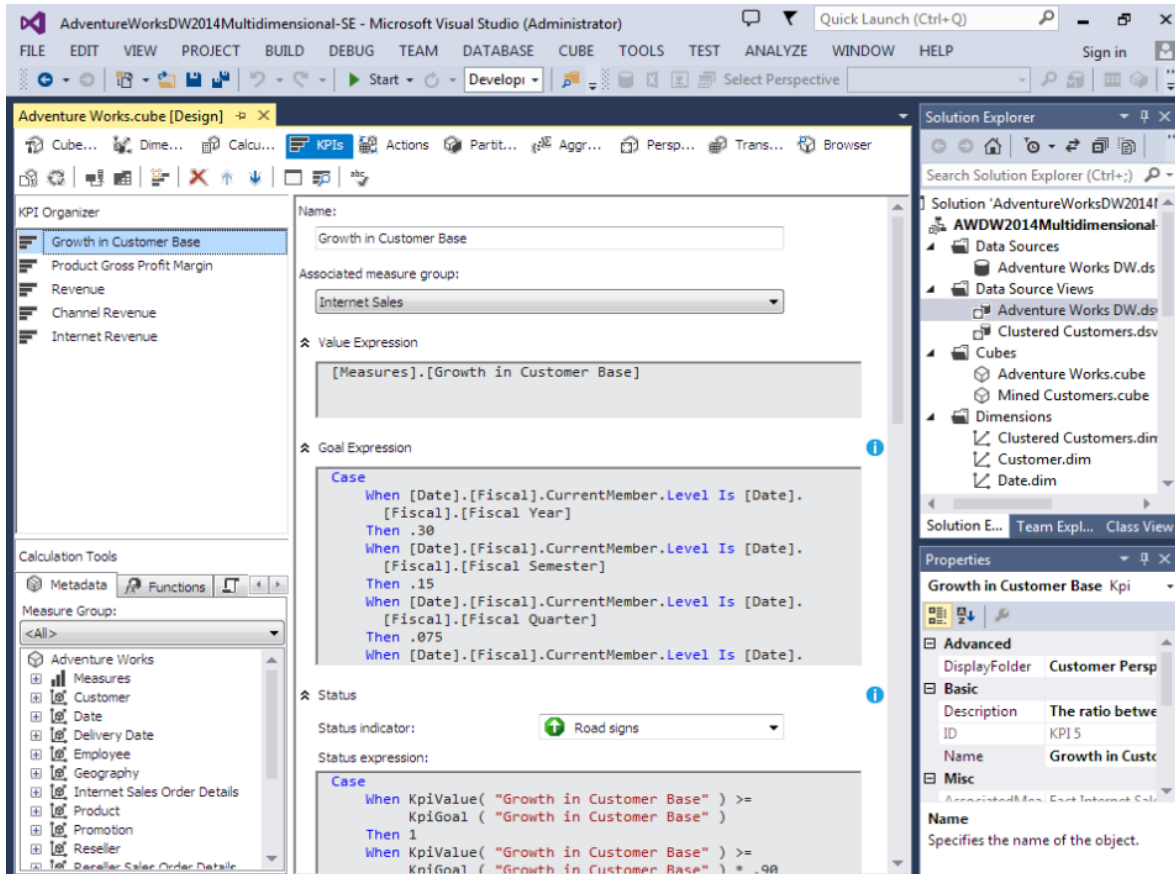
Ένα ακόμη πολύ σημαντικό κομμάτι του OLAP είναι τα KPIs. Τα KPIs είναι αντικείμενα του κύβου τα οποία αντιπροσωπεύουν τους πιο σημαντικούς παράγοντες λειτουργίας μιας επιχείρησης. Είναι πεδία «κλειδί» για την επιτυχία μιας επιχείρησης όπως ποσοστά ή στόχοι, οικονομικοί και μη, που παρακολουθούμε συνεχώς στην επιχειρηματική λογική μιας εταιρίας. Τα KPIs μοιάζουν με τις μετρικές ενός κύβου. Τα KPIs είναι στοιχεία του Business Intelligence μιας επιχείρησης. Τα κύρια στοιχεία ενός KPI είναι :

- **Value** είναι η πραγματική τιμή μπορεί να είναι μια μετρική ή ένα calculation.
- **Goal** είναι ο στόχος που θέλουμε να επιτύχουμε σε μια τιμή.
- **Status** είναι το αποτέλεσμα της σύγκρισης της τιμής με τον επιθυμητό στόχο και συνήθως είναι μια γραφική παράσταση.
- **Trend** είναι η τάση για μια συγκεκριμένη τιμή. Η τάση να πλησιάσει το στόχο ή να απομακρυνθεί από αυτόν. Το trend επίσης μπορεί να είναι μια γραφική παράσταση.

Στο OLAP project Adventure Works στο Design του κύβου τα KPIs είναι το τέταρτο tab όπως βλέπουμε στην εικόνα 69. Στο KPIs organizer βρίσκονται όλα τα KPIs που έχουμε δημιουργήσει για τον κύβο Adventure Works. Στο calculation tools βρίσκονται οι μετρικές του κύβου που μπορούμε να χρησιμοποιήσουμε και τα διάφορα functions της γλώσσας MDX. Τα functions είναι παρόμοια με τα functions του SQL Server για να συντάξουμε ένα ερώτημα και η λογική είναι η ίδια στην γλώσσα MDX.

Το πρώτο πράγμα σε ένα KPI είναι να του δώσουμε ένα όνομα στο Textbox «Name». Στο associated measure group συσχετίζουμε το KPI με την μετρική value που θα χρησιμοποιήσει. Στο KPI το value expression είναι μια μετρική που μπορεί να προέρχεται από ένα πεδίο του πίνακα γεγονότων που φορτώθηκε από την αποθήκη δεδομένων ή από ένα calculated member που δημιουργήσαμε εμείς.

Στο KPI Growth In Customer Base το Value είναι ένα calculated member που υπολογίζει την ανάπτυξη του πελατολογίου της επιχείρησης μέσα στο χρόνο.



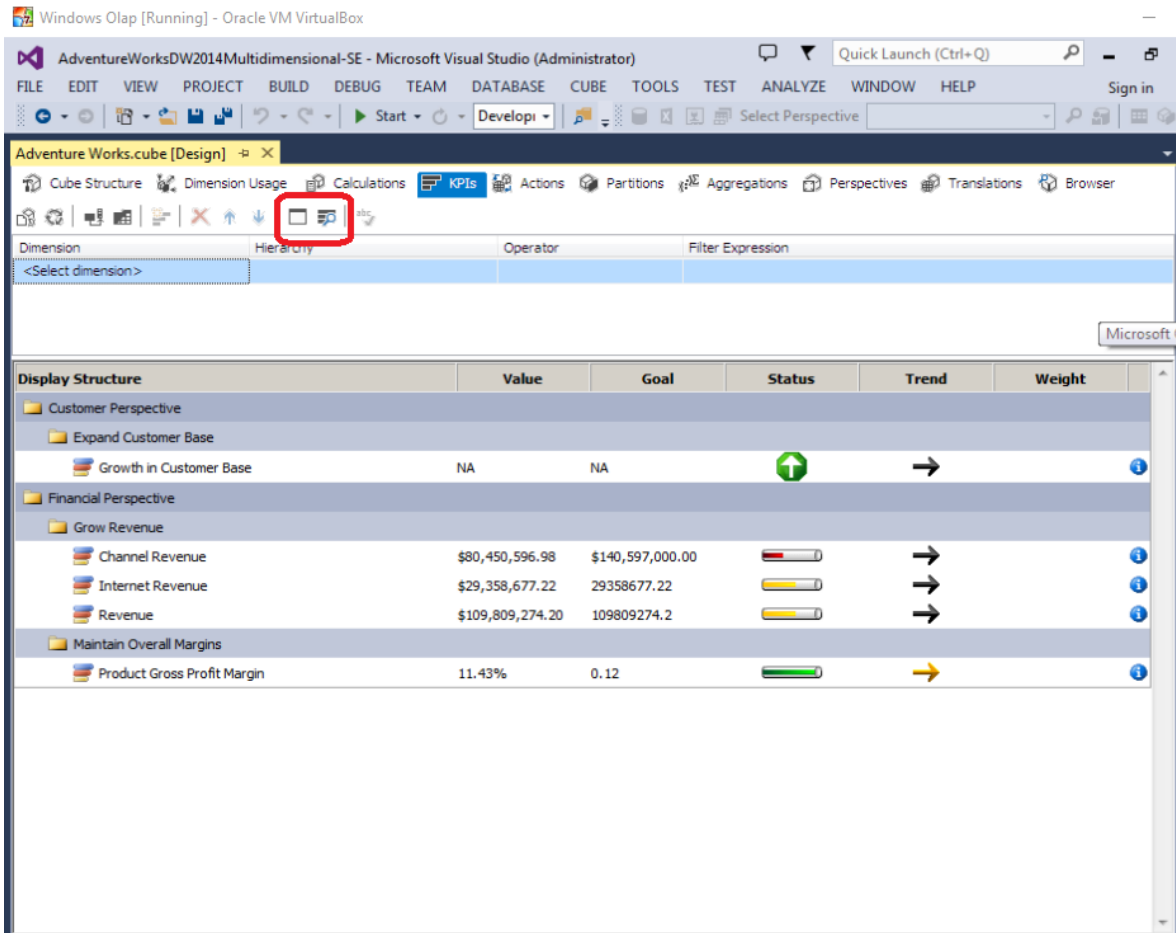
Εικόνα 69 "KPIs"

Το value expression στην περίπτωση μας είναι ένα case σε MDX το οποίο ορίζει πόσο πρέπει να είναι η ανάπτυξη του πελατολογίου σε σχέση με την περίοδο που διανύουμε. Για παράδειγμα στο πρώτο case ορίζουμε ότι η ανάπτυξη του πελατολογίου σε ένα χρόνο πρέπει να είναι 30% και στο δεύτερο case ότι η ανάπτυξη του πελατολογίου στο εξάμηνο πρέπει να είναι 15%. Με αυτό τον τρόπο ορίσαμε τους στόχους για το συγκεκριμένο KPI.

Στο Status Expression ορίζουμε σε MDX να μας επιστρέψει το Status Indicator που επιλέξαμε πιο πάνω αν εκπληρωθούν τα κριτήρια που θέλουμε. Αναλύοντας το expression βλέπουμε ότι αν το KPI value είναι μεγαλύτερο του KPI Goal μας επιστρέφει «1». Μεταφράζοντας το αυτό σημαίνει ότι αν η ανάπτυξη του πελατολογίου στο πεδίο του πίνακα είναι μεγαλύτερη από την ανάπτυξη του πελατολογίου που ορίσαμε στο στόχο εμφανίσει το Indicator πράσινο βελάκι προς τα πάνω. Αν η ανάπτυξη είναι περίπου ίση με το στόχο θα εμφανίσει μια κίτρινη γραμμή και αν η ανάπτυξη πολύ μικρότερη θα εμφανίσει ένα κόκκινο βελάκι προς τα κάτω.

Το trend στο παράδειγμα μας υπολογίζεται από την ανάπτυξη του πελατολογίου τα προηγούμενα. Το πώς κινήθηκε δηλαδή το πελατολόγιο τα προηγούμενα έτη θα χαρακτηρίσει τι τάση υπάρχει για να αναπτυχθεί.

Αφού έχουμε δημιουργήσει το KPI σε αυτό το σημείο μπορούμε να το παρακολουθήσουμε. Στην εικόνα 70 βλέπουμε το Browser View του KPI.

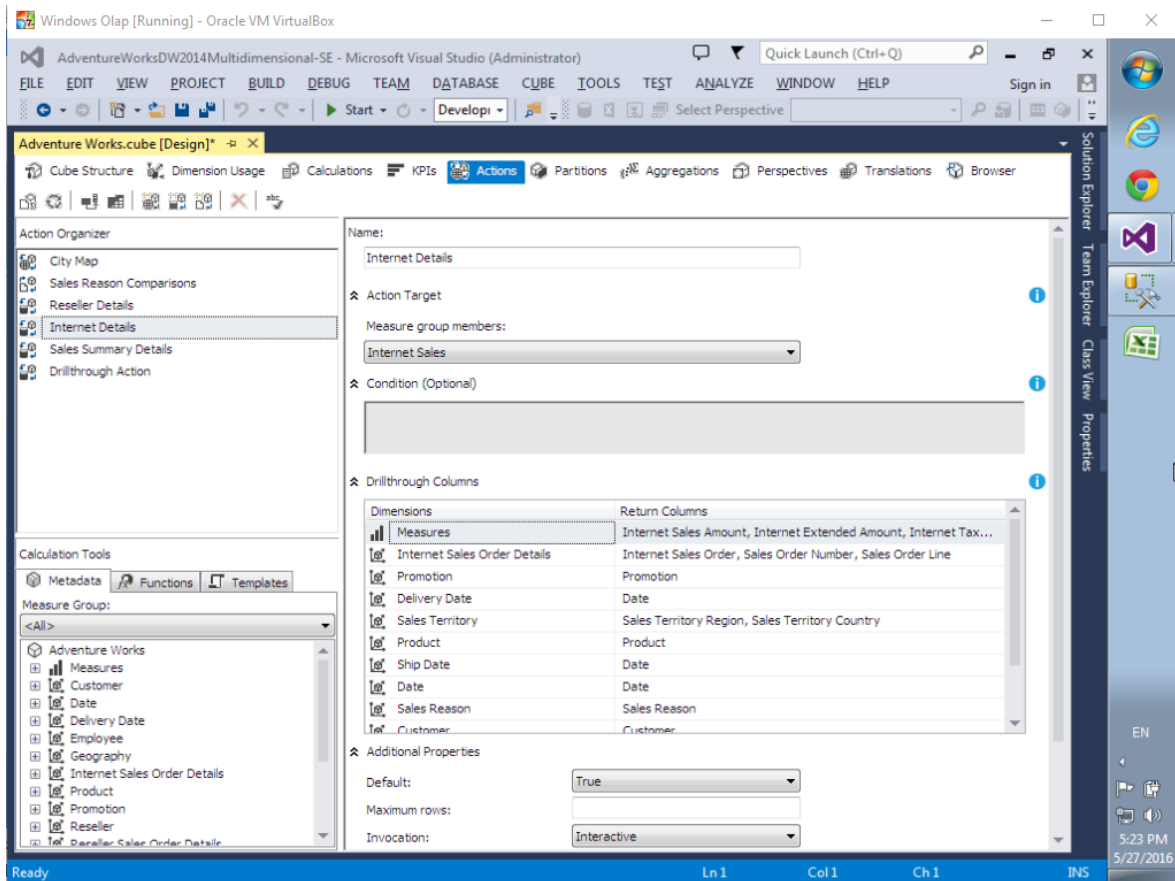


Εικόνα 70 "KPIs Browser View"

Με κόκκινο χρώμα έχουμε κυκλώσει τα κουμπιά Form View και Browser View. Το Form View φαίνεται στην εικόνα 69 που αναλύσαμε προηγουμένως και το Browser View στην εικόνα 70. Στο Browser View βλέπουμε όλα τα KPIs και την ανάλυση τους σε σχέση με το status και το trend. Εδώ είναι σημαντικό να υπογραμμίσουμε ότι ένα KPI μπορεί να αποτελείται από άλλα KPIs τα οποία δίνουν αποτελέσματα ξεχωριστά και στην συνέχεια συνδυάζονται το καθένα με ξεχωριστή βαρύτητα για να παράγουν το τελικό KPI.

### 3.7.3 SSAS Actions

Μία πολύ σημαντική λειτουργία της τεχνολογίας OLAP και των κύβων είναι τα Actions. Στο Design του κύβου στο Visual Studio τα Actions βρίσκονται στο πέμπτο tab μετά τα KPIs. Στην εικόνα 71 βλέπουμε το menu Actions στο Visual Studio στο project του Adventure Works.



Εικόνα 71 "Visual Studio Cube Actions"

Σε αυτό το σημείο πρέπει να αναφέρουμε κάποια περισσότερα πράγματα για τα Actions από αυτά που είδαμε στην θεωρία του κεφαλαίου 2.2. Τα Actions ή operations είναι πρόσθετη λειτουργικότητα την οποία μπορούμε να προσθέσουμε στον κύβο που έχουμε δημιουργήσει. Με την βοήθεια των actions μπορούμε επίσης να δέσουμε τον κύβο με άλλα προγράμματα στα οποία μπορούμε να τον αναλύσουμε όπως reporting Services.

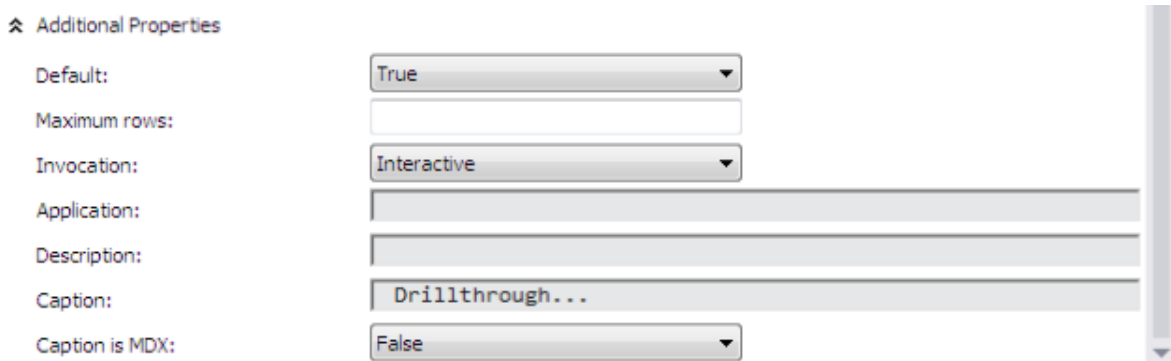
Τα actions συσχετίζονται με τα εξής :

- Ιεραρχίες (Hierarchy)
- Επίπεδα (Level)
- Ιδιότητες (Attribute)
- Μέλη (Members)
- Κελιά (Cells)
- Κύβους (Cubes)

Στην εικόνα 72 βλέπουμε ένα παράδειγμα ενός drill through action με όνομα Internet details. Η μετρική που κάνει target είναι ο πίνακας γεγονότων internet sales. Μέσα στο condition μπορούμε αν θέλουμε με γλώσσα MDX να εξαιρέσουμε μετρικές που δεν θέλουμε από τον πίνακα γεγονότων που κάναμε target. Στο panel drill through columns επιλέγουμε τους πίνακες που επιθυμούμε οι οποίοι

μπορεί να είναι είτε dimensions του κύβου είτε μετρικές του κύβου και στο return columns τις στήλες των πινάκων που θέλουμε να συμμετάσχουν στο action.

Τέλος στο additional properties που βρίσκεται στο κάτω μέρος του action και βλέπουμε στην εικόνα 72 μπορούμε να κάνουμε κάποιες ρυθμίσεις που αφορούν το action.



Additional Properties

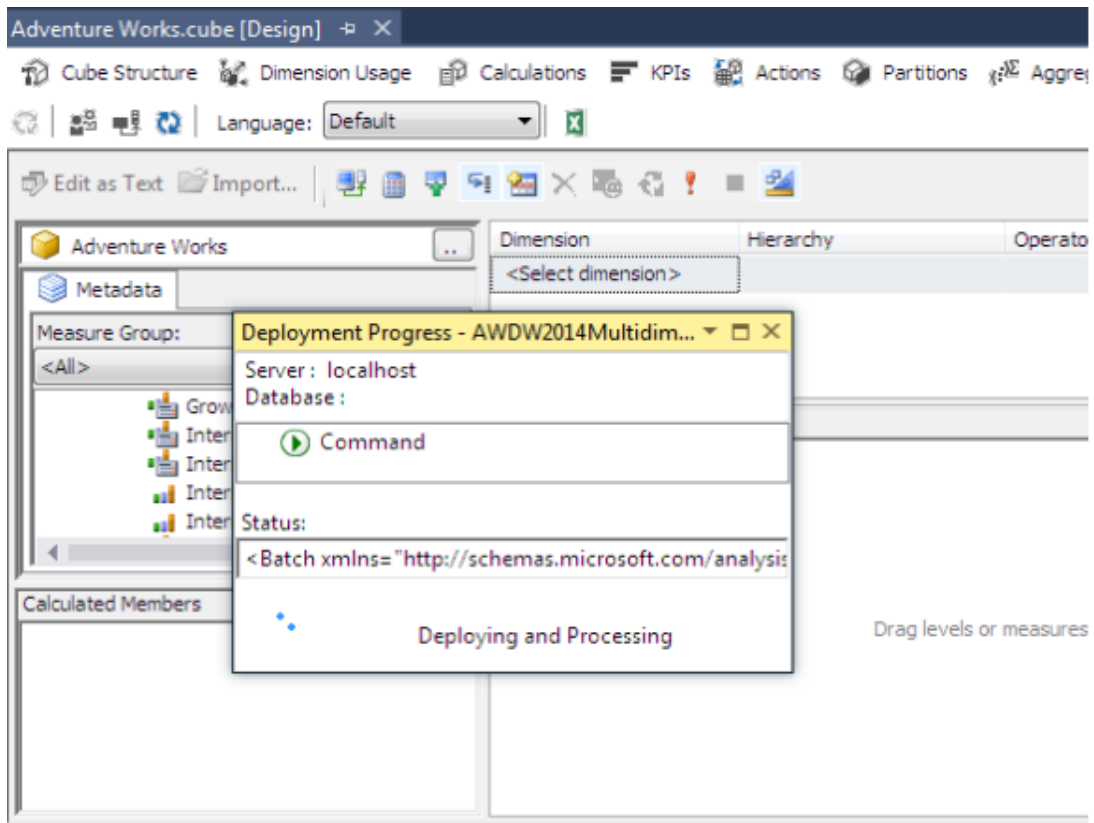
Default:	True
Maximum rows:	
Invocation:	Interactive
Application:	
Description:	
Caption:	Drillthrough...
Caption is MDX:	False

Εικόνα 72 "Action Additional Properties"

Το default δεν είναι κάποια σημαντική λειτουργία και αφορά την διαδικασία της αναζήτησης των δεδομένων κατά την εκτέλεση του action, το αφήνουμε στην τιμή που έχει. Το maximum rows ορίζει το όριο στις γραμμές που επιστρέφουν από το query. Το invocation είναι για την συμπεριφορά όταν καλείτε το action από κάποιον client. Στο application ονομάζουμε την εφαρμογή που θα το καλεί, στο παράδειγμα μας δεν το χρειαζόμαστε. Στο description αν θέλουμε γράφουμε μια περιγραφή του action. Το caption αντικαθιστά το όνομα αν το επιθυμούμε πάλι για εξωτερική χρήση του action και τέλος το «caption is MDX» το αφήνουμε πάντα false.

Σε αυτό το σημείο έχουμε δημιουργήσει ένα drill through action το οποίο μπορούμε να εφαρμόσουμε στους πίνακες και στα πεδία που του ορίσαμε. Αφού κάνουμε deploy το Solution η λειτουργία αυτή προστίθεται στον κύβο για ανάλυση.





Εικόνα 73 "Deploying Solution"

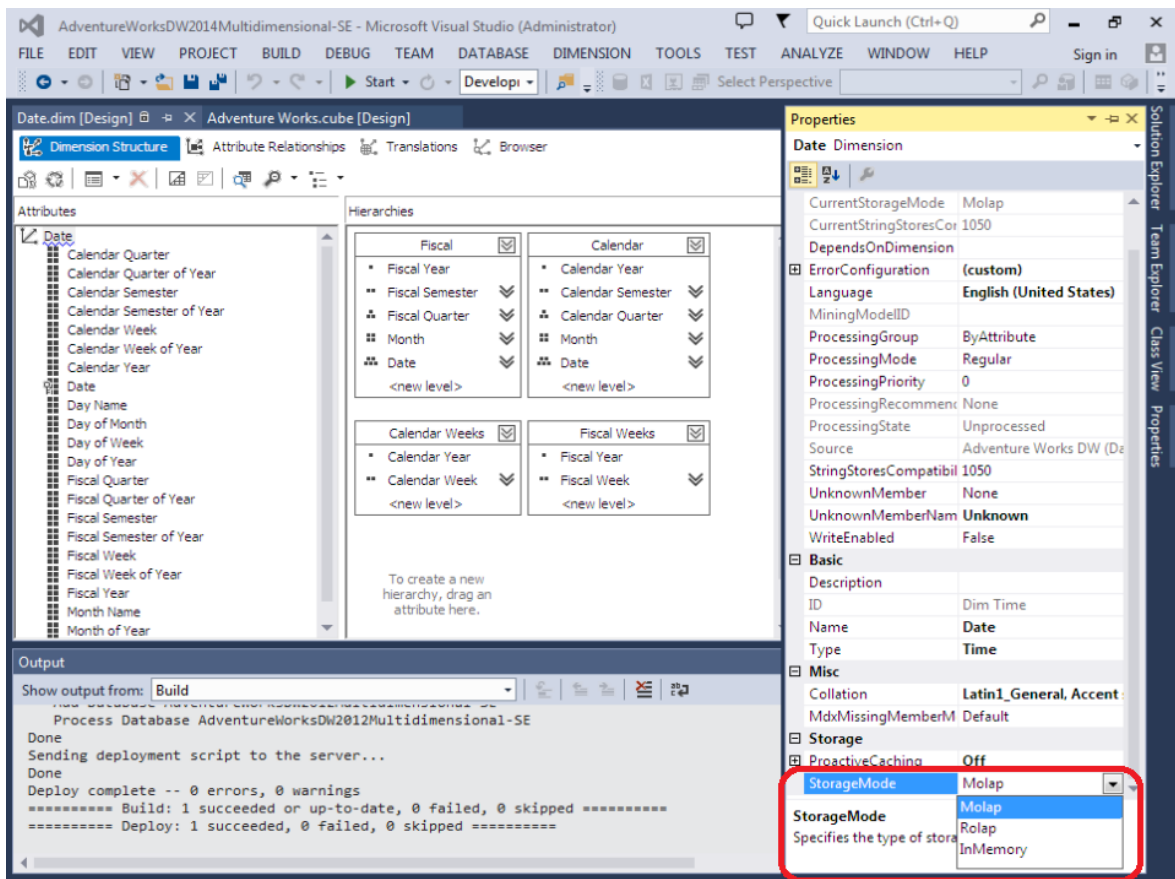
Στην εικόνα 73 βλέπουμε το παράθυρο που κάνει deploy το Solution στον Analysis Server που έχουμε τοπικά στο μηχάνημα.

### 3.7.4 SSAS Storage Options και Partitions

Σε αυτό το κεφάλαιο θα δούμε και θα μελετήσουμε την αποθήκευση των δεδομένων ενός OLAP project στο Visual Studio. Στο κεφάλαιο 2.5 παρουσιάσαμε το θεωρητικό μέρος των τρόπων αποθήκευσης που περιλαμβάνουν τρεις τρόπους MOLAP, ROLAP και HOLAP. Το MOLAP συνήθως είναι η default επιλογή όταν αποθηκεύουμε πίνακες γεγονότων και διαστάσεις σε ένα κύβο. Οι διαστάσεις μπορούν να αποθηκευθούν μόνο σαν MOLAP ή ROLAP ενώ οι κύβοι μπορούν να χρησιμοποιήσουν και τις τρεις δομές.

Ένα σημαντικό πράγμα που πρέπει να αναφέρουμε είναι ότι όταν ένας κύβος γίνεται process και αποθηκεύεται σε MOLAP δημιουργείται ένα snapshot του κύβου το οποίο χρησιμοποιείται για όλα τα query που θα τρέξουν στον κύβο και οι σχεσιακές βάσεις δεδομένων και αποθήκες δεδομένων που προμήθευσαν τα δεδομένα δεν χρησιμοποιούνται. Συνεπώς αν αλλάξουν τα δεδομένα της αποθήκης και θέλουμε να τα επεξεργαστούμε στο κύβο, ο κύβος πρέπει να γίνει reprocess.

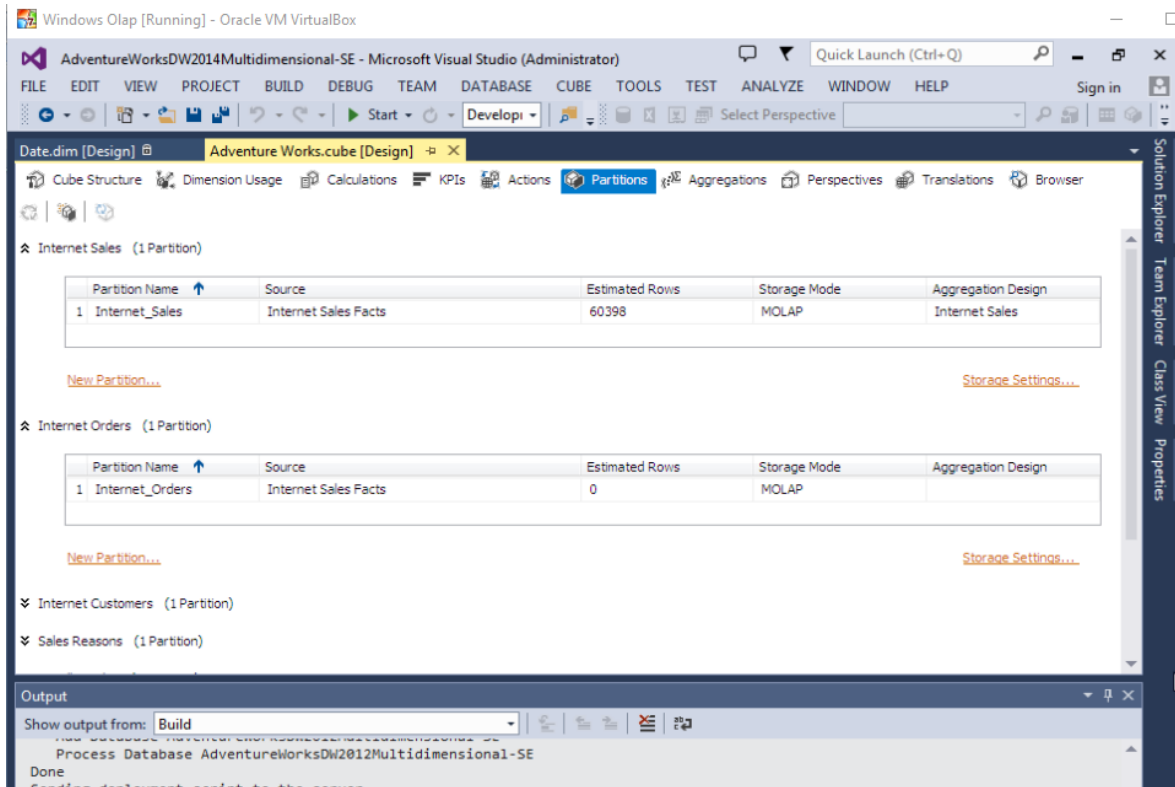
Στην εικόνα 74 βλέπουμε τα properties της διάστασης Date στο project Adventure Works στο Visual Studio.



Εικόνα 74 "Dimension Storage Options"

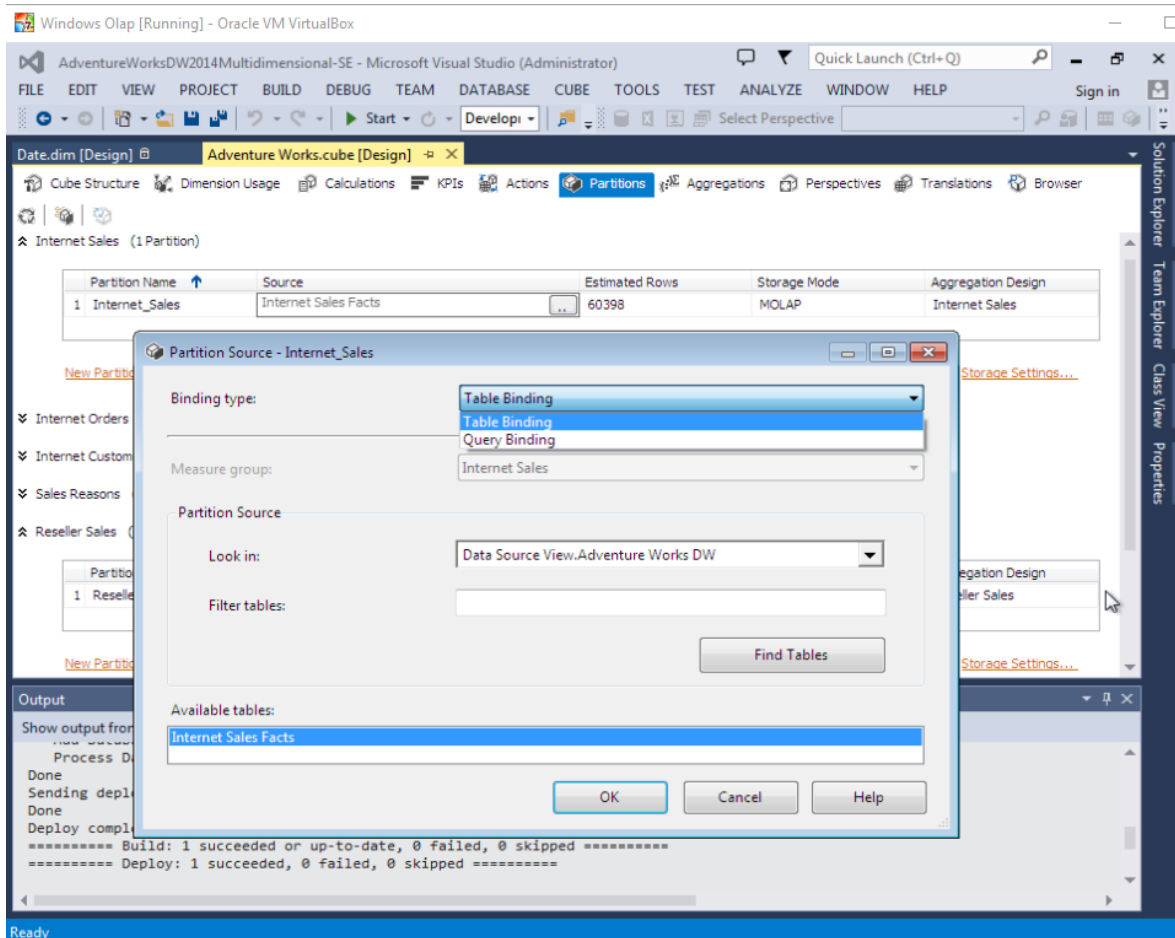
Η default αποθήκευση είναι MOLAP και μπορούμε να επιλέξουμε αν θέλουμε να αποθηκευτεί σε ROLAP ή In Memory. Μπορούμε αν θέλουμε κάποιες διαστάσεις να τις αποθηκεύσουμε σε MOLAP και άλλες διαστάσεις να τις αποθηκεύσουμε σε ROLAP μορφή.

Η αποθήκευση του κύβου ορίζεται από το έκτο tab στο Design του κύβου που ονομάζεται partitions. Στην εικόνα 75 βλέπουμε τα partitions του κύβου Adventure Works.



Εικόνα 75 "Adventure Works Partitions"

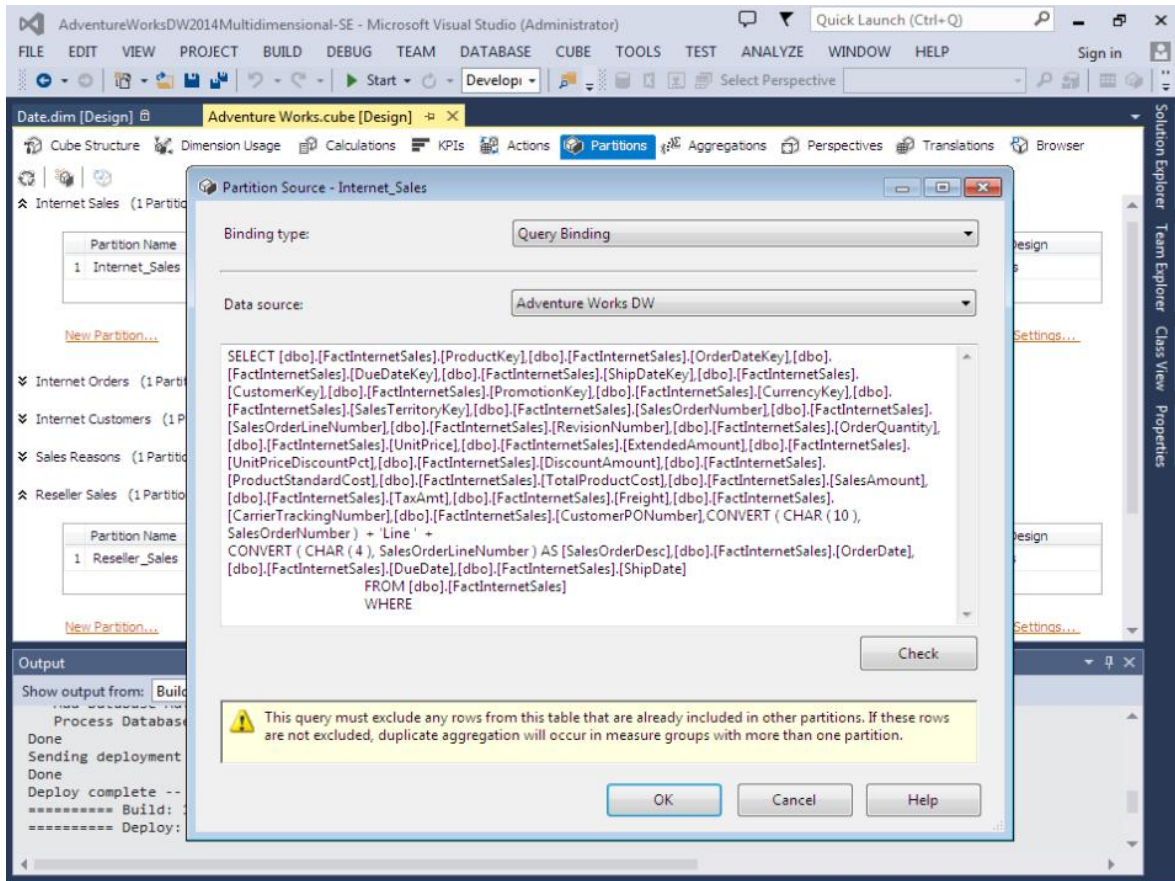
Τα partitions μας παρέχουν την δυνατότητα να σπάσουμε τον κύβο σε διαφορετικά αρχεία όταν γίνεται η αποθήκευση του. Για παράδειγμα μπορούμε να δημιουργήσουμε διαφορετικά partitions για κάθε πίνακα γεγονότων. Επίσης μπορούμε να έχουμε partitions σχετικά με το χρόνο, ώστε κάθε partition για παράδειγμα να αποθηκεύει μια συγκεκριμένη χρονιά με τα δεδομένα της επιχείρησης. Αυτή η τεχνική μπορεί να βοηθήσει σημαντικά το performance του κύβου. Στην εικόνα 75 βλέπουμε το partition Internet Sales και το Source του είναι ο πίνακας γεγονότων Internet Sales Facts. Κάνοντας click πάνω στο Internet Sales Facts μας ανοίγει το παράθυρο που βλέπουμε στην εικόνα 76.



Εικόνα 76 "Internet Sales Partition Source"

Σαν πηγή δεδομένων για το partition έχουμε δύο επιλογές και μπορεί να είναι Table Binding ή Query Binding. Στο table binding ορίζουμε ένα data Source ή data View και τέλος ορίζουμε τον πίνακα που θέλουμε.

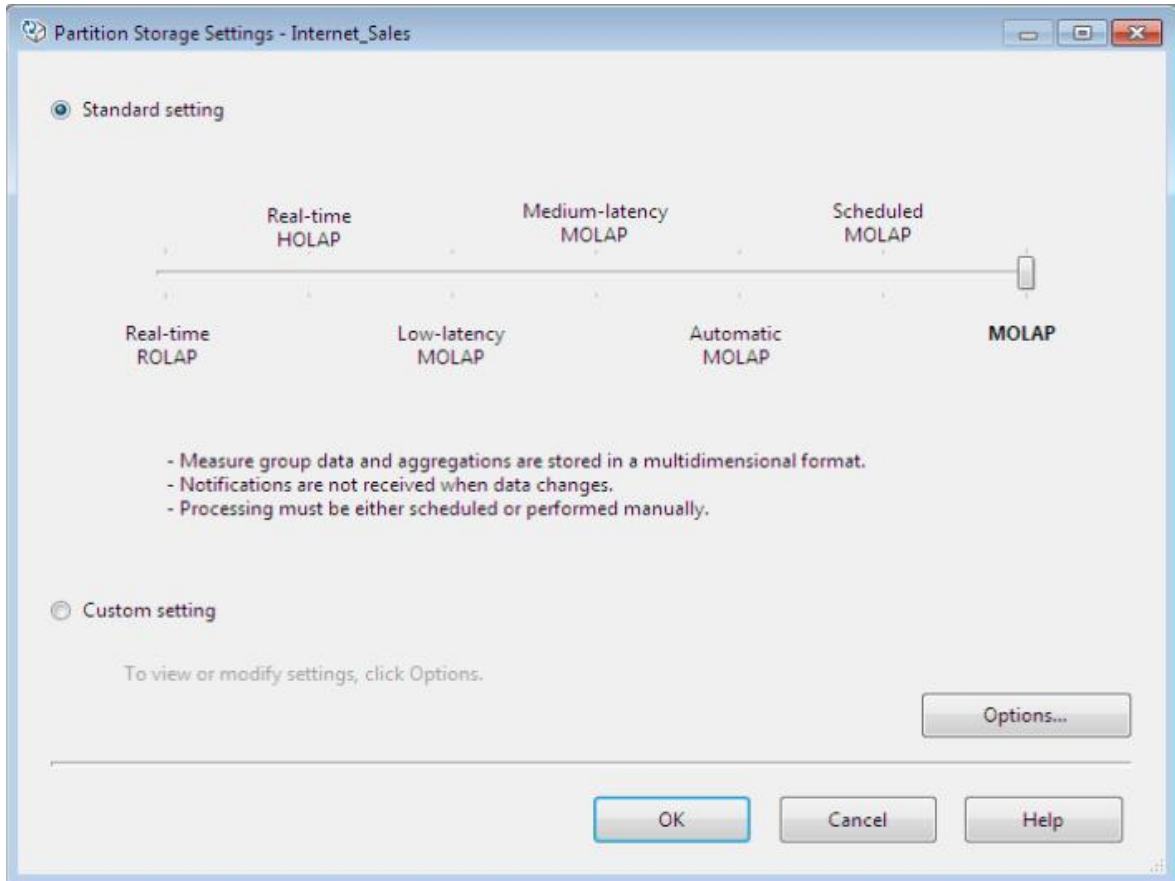
Στο query binding που βλέπουμε στην εικόνα 77 ορίζουμε το data Source και από κάτω μας δημιουργεί ένα select Statement αφήνοντας το Where στο τέλος για να επανασχεδιάσουμε ή να συμπληρώσουμε εμείς το query για το πια δεδομένα θέλουμε να φορτώσουμε στο partition. Με το κουμπί check ελέγχουμε την σύνταξη του query.



Εικόνα 77 "Internet Sales Partition Query Binding"

Για παράδειγμα στο υπάρχον partition Internet Sales θα μπορούσαμε να επιλέξουμε Query binding και στο Where clause του Select statement να επιλέξουμε το πεδίο Date να είναι ανάμεσα στις ημερομηνίες 01/01/2015 και 31/12/2015. Με αυτό τον τρόπο θα δημιουργήσουμε ένα partition με τις πωλήσεις του Internet μόνο για το 2015 το οποίο θα είναι υποσύνολο του Internet Sales. Να υπενθυμίσουμε σε αυτό το σημείο ότι ο κύβος πάντα μετά από μια αλλαγή χρειάζεται deploy και process για να φορτώσει τις αλλαγές.

Για να ορίσουμε τον τρόπο αποθήκευσης ενός partition πηγαίνουμε στο Storage Settings του κάθε partition που έχουμε δημιουργήσει που φαίνεται στην εικόνα 75. Αφού το επιλέξουμε μας ανοίγει το παράθυρο partition storage setting που βλέπουμε στην εικόνα 78.



Εικόνα 78 "Partition Storage Settings"

Σε αυτό το σημείο με την χρήση του slide scale ορίζουμε τον τρόπο αποθήκευσης του partition από ROLAP μέχρι MOLAP. Ενδεικτικά να αναφέρουμε ότι το ROLAP προσφέρει real time δεδομένα με αρκετά μειονεκτήματα στο performance. Στην αντίθετη πλευρά το MOLAP μπορεί να μην παρέχει δυνατότητα ανάλυσης real time αλλά είναι πολύ πιο αποδοτικό από το ROLAP.

Συμπερασματικά πολλές εταιρίες μπορεί εσφαλμένα να θεωρούν ότι χρειάζονται να αναλύουν real time δεδομένα για να χρησιμοποιήσουμε ROLAP ενώ στην πραγματικότητα μπορούμε να εφαρμόσουμε την ίδια επιχειρηματική ευφυΐα με MOLAP χωρίς το μικρό latency στα δεδομένα να μας επηρεάσει αρνητικά. Το MOLAP είναι το επικρατέστερο αυτή την στιγμή στην αγορά ανάμεσα στις τρεις δομές.

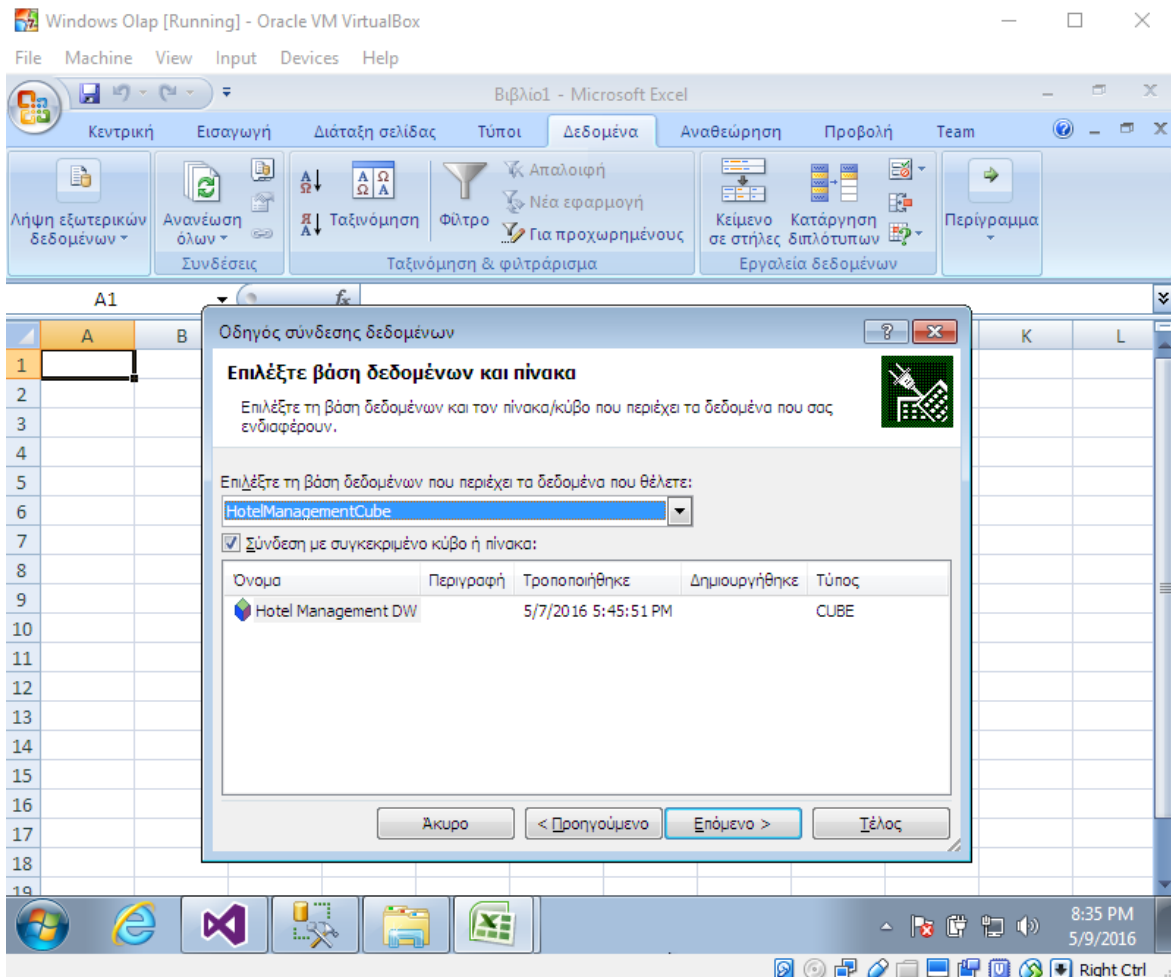
### 3.8 Ο κύβος OLAP στο Microsoft Excel

Ένας κύβος δημιουργημένος στο Visual Studio μπορεί να επεξεργαστεί και στο περιβάλλον του Microsoft Excel χωρίς να είναι απαραίτητη η χρήση κάποιου plugin. Το Excel παρέχει όλες τις λειτουργίες που χρειαζόμαστε για να φορτώσουμε ένα κύβο, να τον αναλύσουμε και να δημιουργήσουμε High Value πληροφορία

από τα δεδομένα του. Με αυτό τον τρόπο παρέχεται η δυνατότητα και η ευελιξία να μελετηθεί ένας πολυδιάστατος κύβος σε ένα προϊόν συνηθισμένο στις επιχειρήσεις όπως το excel.

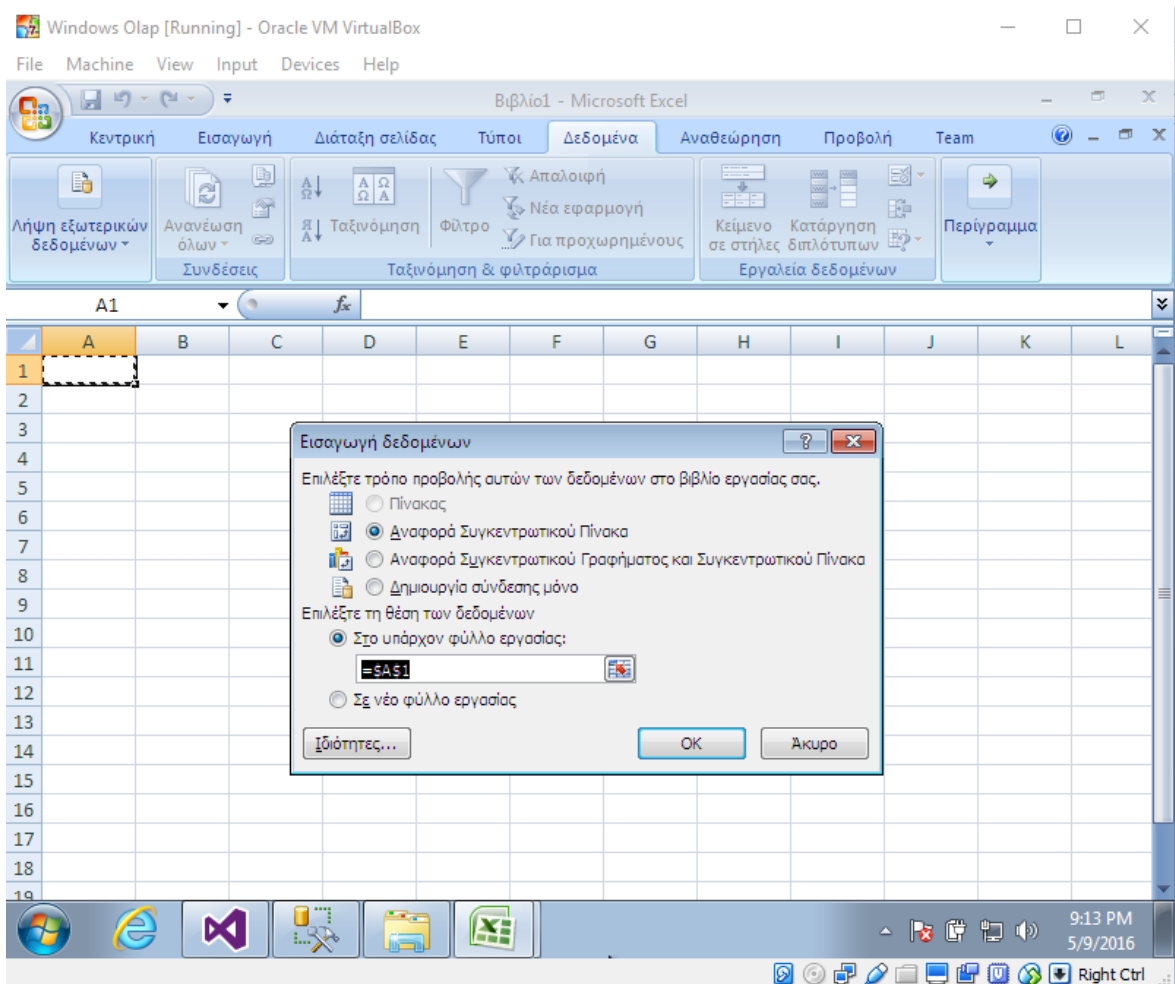
Στην πτυχιακή αυτή χρησιμοποιούμε το Office 2007, στα Office 2010, 2013 και μετέπειτα προστίθενται και άλλες λειτουργίες που βελτιώνουν ακόμη περισσότερο την χρήση του Excel. Παρότι δεν χρειαζόμαστε κάποιο plug in για να αρχίσουμε να αναλύουμε τον κύβο ευθύς αμέσως έχουμε εγκαταστήσει το OLAP PivotTable extensions από το Codeplex που ενισχύει την λειτουργικότητα του Excel.

Ανοίγοντας το excel πηγαίνουμε στο μενού δεδομένα - στην λήψη εξωτερικών δεδομένων - από άλλες προελεύσεις - από υπηρεσίες ανάλυσης. Μας ανοίγει το παράθυρο για να συνδεθούμε στον server που έχει τον κύβο όπως φαίνεται στην εικόνα 79. Στην περίπτωση μας είναι OLAP-PC και η βάση που περιέχει τον κύβο Hotel Management Cube.



Εικόνα 79 "Excel σύνδεση στον Analysis Server"

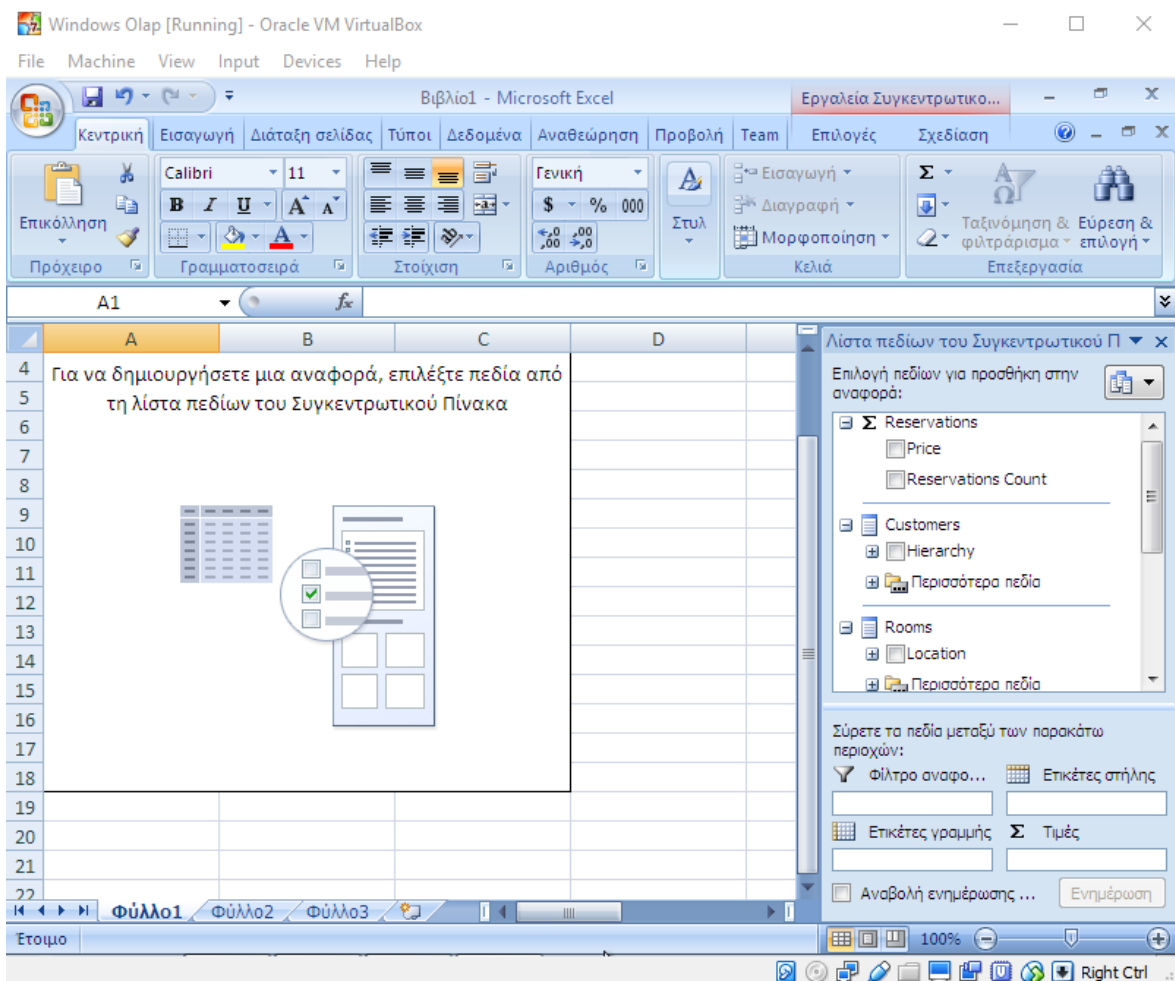
Στην επόμενη οθόνη εικόνα 80 επιλέγουμε το πώς θέλουμε να μας παρουσιαστούν τα δεδομένα.



Εικόνα 80 "Excel Παρουσίαση δεδομένων κύβου – Αναφορά συγκεντρωτικού πίνακα"

Επιλέγουμε την default παρουσίαση Αναφορά Συγκεντρωτικού πίνακα (Table Pivot Report). Ο κύβος τώρα έχει φορτωθεί στο excel και είναι έτοιμος να μελετηθεί, να δημιουργηθούν reports και να παραχθεί πληροφορία πάνω στα δεδομένα του.





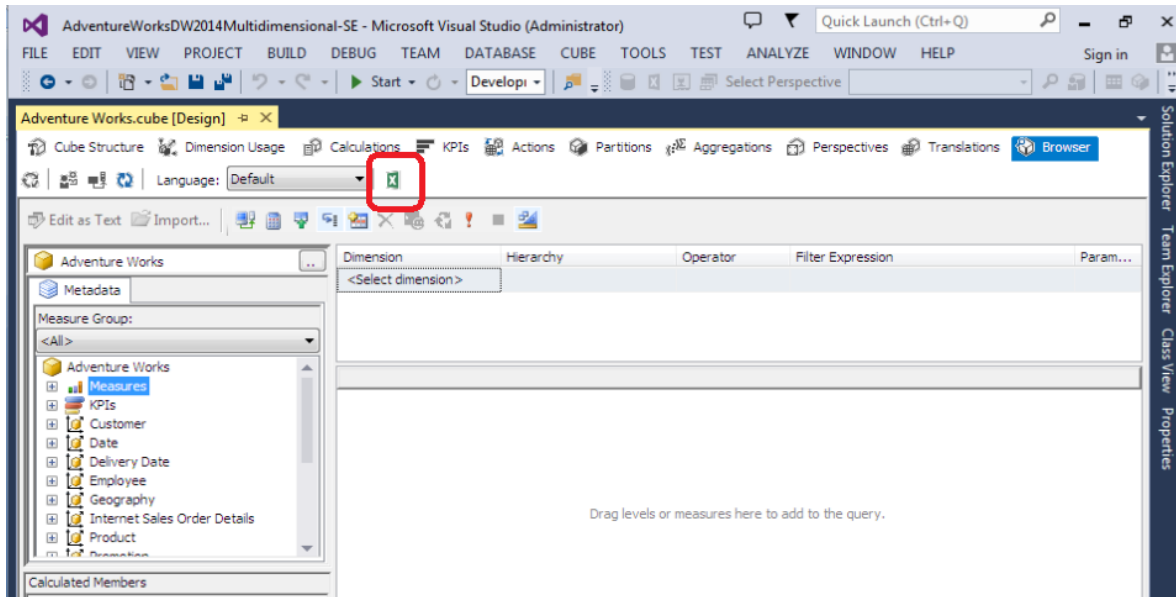
Εικόνα 81 "Excel OLAP"

Τέλος στην εικόνα 81 βλέπουμε το περιβάλλον στο Excel που μπορούμε να κάνουμε αναλυτική επεξεργασία του κύβου Hotel Management με απλό check και drag and drop σε μετρικές και διαστάσεις χρησιμοποιώντας όλες τις τεχνικές επεξεργασίας του κύβου όπως να επιλέξουμε στήλες και γραμμές, να επιλέξουμε τα values που μας ενδιαφέρουν καθώς και να κάνουμε filtering στους πίνακες που επιθυμούμε. Παρακάτω θα παρουσιάσουμε την ίδια οθόνη και για τα στοιχεία του project Adventure Works.

Ο πρώτος τρόπος που είδαμε για το πώς να φορτώσουμε στο Excel ένα κύβος είναι αρκετά εύκολος και χρηστικός για clients της επιχείρησης. Οποιοσδήποτε επιθυμούμε στην επιχείρηση που έχει πρόσβαση στην βάση δεδομένων του κύβου μέσω του δικτύου μπορεί να συνδεθεί στην βάση, να αναλύσει και να μελετήσει τον κύβος που έχει δημιουργήσει κάποιος Business Intelligence Officer και να δημιουργήσει πολύτιμα reports για όποιο τμήμα της εταιρίας χρειαστεί χωρίς να χρειάζεται να ξέρει τι συμβαίνει κατά το development του OLAP project ή να έχει κάποια ιδέα τι είναι Management Studio ή Visual Studio.

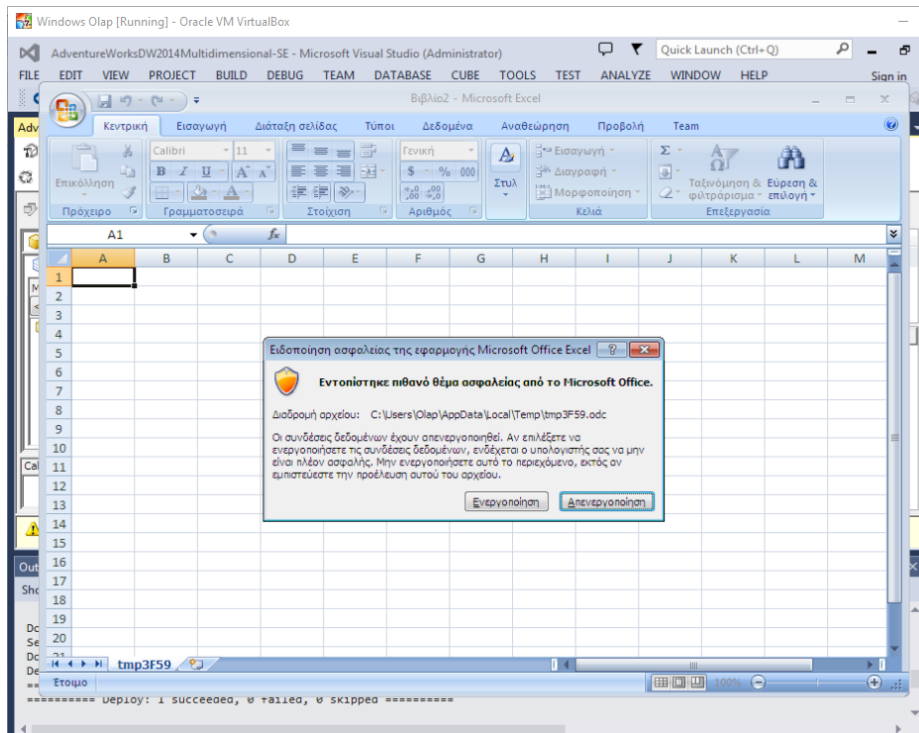
Ο δεύτερος τρόπος που θα δούμε είναι μέσα από το Visual Studio με μια λειτουργία που λύνει τα χέρια στους διάφορους developers και BI officers που

χρειάζεται να δημιουργήσουνε reports στο Excel. Στην εικόνα 82 στο Project Adventure Works στο Visual Studio με κόκκινο χρώμα φαίνεται το κουμπί που λέγεται analyze in Excel. Βρισκόμαστε στο tab Browser του κύβου adventure Works και η λειτουργία του analyze in excel μας επιτρέπει να ανοίξουμε και να φορτώσουμε τον κύβο που έχουμε κάνει deploy κατευθείαν στο excel.



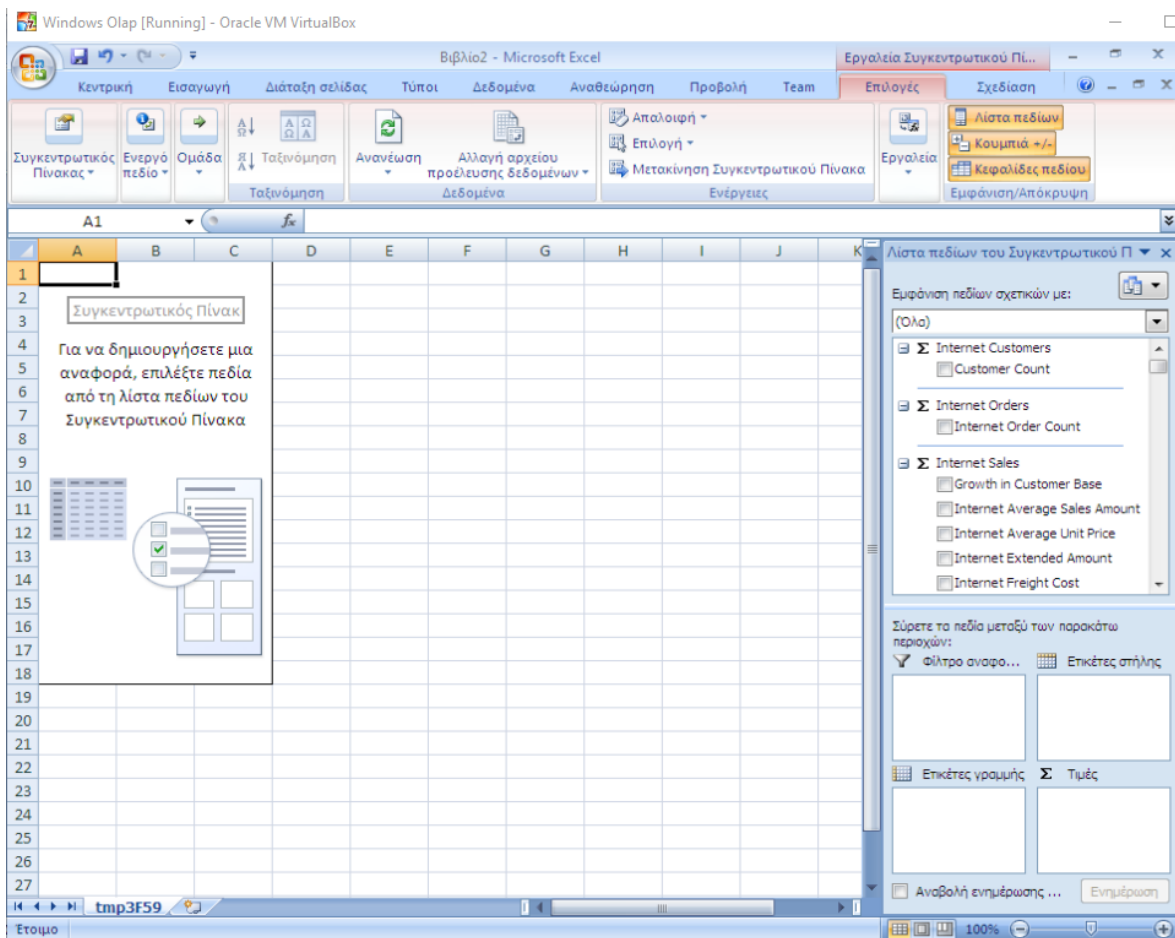
Εικόνα 82 "Analyze In Excel"

Πατώντας το κουμπί το Visual Studio ανοίγει το excel και η πρώτη οθόνη που βλέπουμε στην εικόνα 83 είναι μια ειδοποίηση ασφαλείας για τα δεδομένα που πάει το Visual Studio να φορτώσει στο excel. Επιλέγουμε ενεργοποίηση.



Εικόνα 83 "OLAP In Excel"

Στην επόμενη οθόνη στην εικόνα 84 το Excel έχει φορτώσει τον κύβο του Visual Studio και είμαστε έτοιμοι να κάνουμε οποιαδήποτε ανάλυση θέλουμε.



Εικόνα 84 "OLAP Adventure Works In Excel"

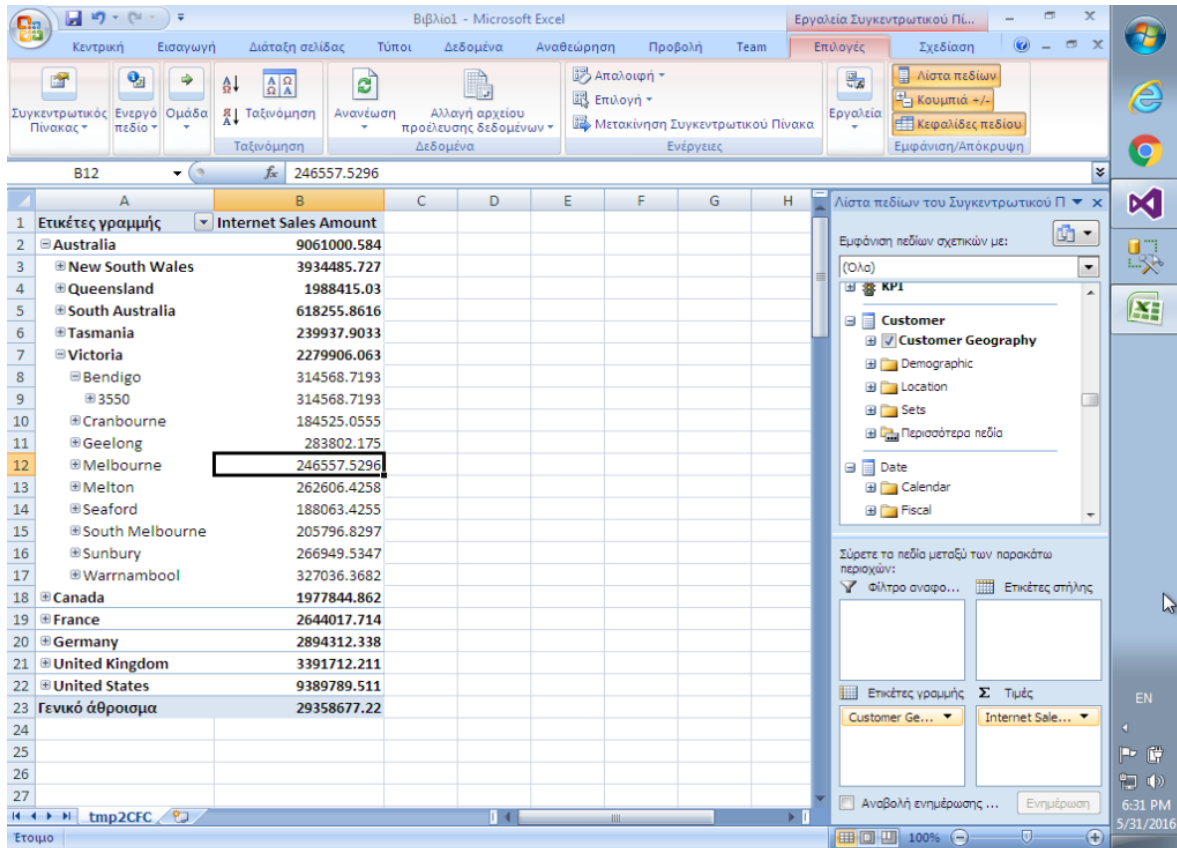
Με αυτό τον τρόπο από το Visual Studio μπορούμε να μεταφερθούμε πολύ απλά στο περιβάλλον του Excel και από εκεί να δημιουργήσουμε reports και να τα δώσουμε σε μια μορφή αρχείου .xls που οποιοσδήποτε θα μπορεί να αναγνωρίσει και διαβάσει.

Στην εικόνα 84 στα δεξιά βλέπουμε όλους τους πίνακες γεγονότων με τα πεδία τους τις μετρικές και τα calculated members. Τα KPIs που έχουμε δημιουργήσει, τις διαστάσεις με τις ιεραρχίες και τα πεδία τους.

Από κάτω είναι όλες οι λειτουργίες, actions ή operations που μπορούμε να εφαρμόσουμε πάνω στο κύβο και τα στοιχεία του, όπως filters, επιλογή rows και columns καθώς και τα values που μας ενδιαφέρουν.

Επιλέγοντας από το PivotTable συγκεκριμένες μετρικές και διαστάσεις με τις ιεραρχίες τους, και κάνοντας filtering για να παρουσιάσουμε δεδομένα, υλοποιούμε όλο το εύρος των τεχνικών dicing, slicing, drill through και roll up.

Για παράδειγμα έστω ότι θέλουμε να δούμε όλες τις πωλήσεις μέσω Internet ανά πελάτη ανά γεωγραφική περιοχή στον κύβο Adventure Works. Στην εικόνα 85 επιλέγουμε την μετρική Internet Sales Amount και από τις διαστάσεις επιλέγουμε την ιεραρχία Customer Geography.



Εικόνα 85 "OLAP Cube Excel Query"

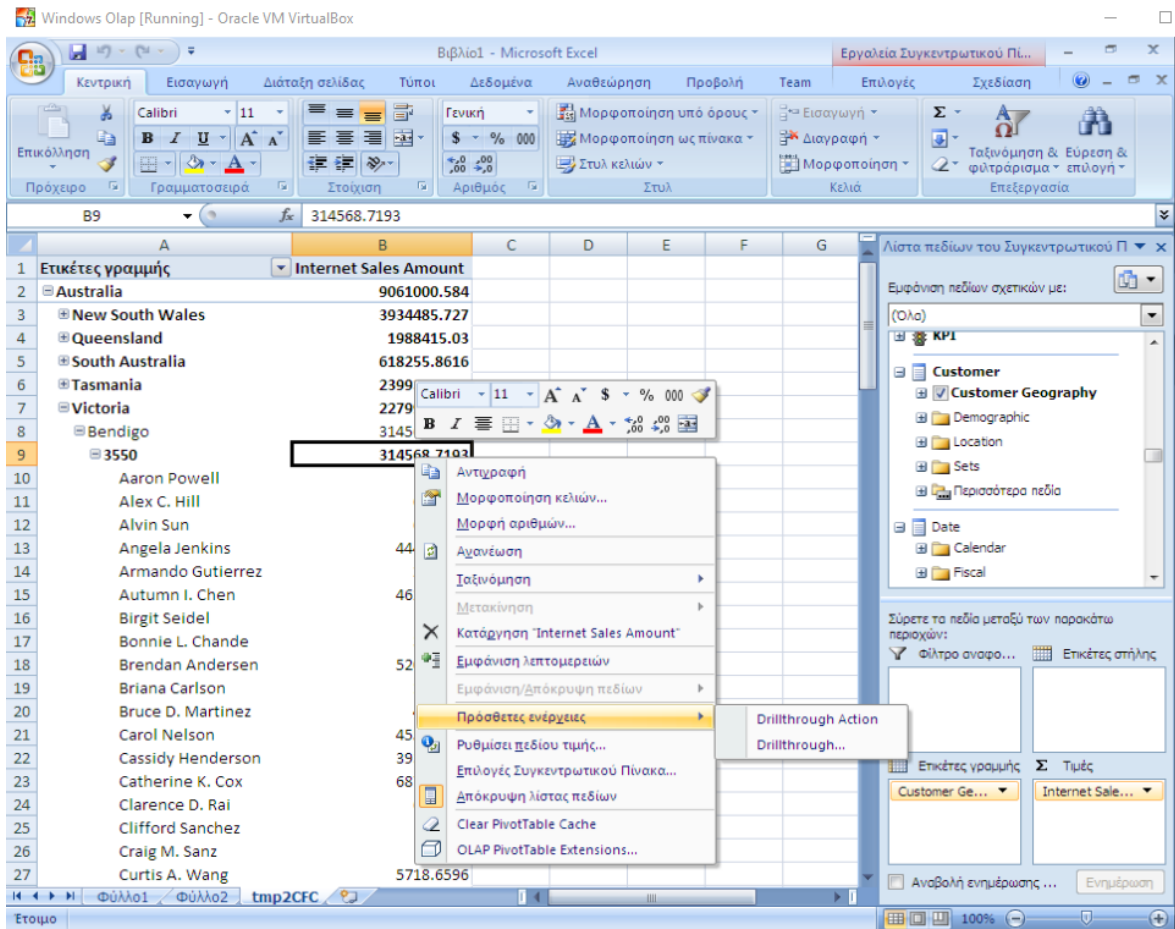
Τα δεδομένα που βλέπουμε έχουν παραχθεί από τον κύβο κάνοντας τον slice και dice καθώς τον περιορίσαμε σε εμφάνιση ενός πεδίου (Internet Sales Amount) σε ένα από τους πίνακες γεγονότων και με την χρήση μιας μόνο διάστασης την Customer Geography.

Στην εικόνα μπορούμε επίσης να παρατηρήσουμε το φίλτρο στο A1 κελί Ετικέτες γραμμής που μας επιτρέπει να επιλέξουμε το βάθος των δεδομένων που παρουσιάζουμε. Κάνοντας click στα πεδία μπορούμε να κάνουμε drill through ή roll up στα δεδομένα για να δούμε τα αποτελέσματα από το επίπεδο χώρας και να φτάσουμε σε επίπεδο λεπτομέρειας μέχρι το όνομα του πελάτη.

Στο σημείο αυτό πρέπει να παρουσιάσουμε μια ακόμη πολύ σημαντική λειτουργία του Excel η οποία μας επιτρέπει actions που έχουν δημιουργηθεί στο Visual Studio να εκτελεστούν στο Excel πάνω στα δεδομένα μας.

Στο κεφάλαιο 3.7.3 δημιουργήσαμε actions στο project Adventure Works στο tab Actions του κύβου. Αυτά μπορεί να ποικίλουν από απλά αυτοματοποιημένα Dice, Slice και drill through, roll up στο κύβο μέχρι πολύπλοκες ενέργειες και

expressions, με συνδυαστικά Queries πάνω σε πολύπλοκη επιχειρηματική λογική. Το excel μας δίνει την δυνατότητα να τρέξουμε αυτά τα actions πάνω στα δεδομένα μας όπως βλέπουμε στην εικόνα 86.



Εικόνα 86 "OLAP Cube Excel Actions"

Στο παράδειγμα των συνολικών πωλήσεων από το Internet μπορούμε να κάνουμε δεξί click στο πεδίο του ποσού και να πάμε στο πρόσθετες ενέργειες. Εκεί θα μας εμφανιστούν όλα actions έχουμε δημιουργήσει στο Visual Studio τα οποία σχετίζονται με την μετρική Internet Sales Amount. Στο παράδειγμα μας έχουμε δύο actions τα οποία κάνουν το μέγιστο drill through στα δεδομένα και τα παρουσιάζουν όσο το δυνατόν πιο λεπτομερειακά. Πατώντας πάνω στο action που επιθυμούμε θα μας ανοίξει ένα καινούριο φύλλο στο Excel που θα περιέχει τα δεδομένα που παρουσιάζονται από την εκτέλεση του Action.

Αυτή η λειτουργία μας επιτρέπει να δημιουργήσουμε αυτοματοποιημένες διαδικασίες επί του κύβου και των δεδομένων, τις οποίες οι τελικοί χρήστες του OLAP τις επιχείρησης θα μπορούν απλά με το πάτημα ενός κουμπιού να τις τρέξουν, απλοποιώντας ακόμη περισσότερο την ανάλυση των δεδομένων για ένα χρήστη που είναι «agnostic» με το background της τεχνολογίας OLAP.

### 3.9 Εκπαιδευτική Λίστα με Video Tutorials

Στην πτυχιακή αυτή δημιουργήσαμε ένα έγγραφο word με μία λίστα από διάφορα video tutorials που υπάρχουν κυρίως στην ιστοσελίδα YouTube. Η λίστα αυτή έχει σκοπό, με την παρακολούθηση των διάφορων video, να εκπαιδεύσει τον αναγνώστη πάνω στο ευρύτερο φάσμα της τεχνολογίας OLAP και συγκεκριμένα με την χρήση προϊόντων της Microsoft. Ενδεικτικά να αναφέρουμε τους τομείς που παρουσιάζονται στα βίντεο αυτά.

- **Data Warehouse:** Εδώ παρουσιάζεται η έννοια του Data Warehousing, η αρχιτεκτονική του, γιατί είναι απαραίτητο, πως χρησιμοποιείτε και τα πλεονεκτήματα του σε σχέση με την παραδοσιακή βάση δεδομένων.
- **ETL:** Αναλύεται τι είναι το ETL και με πιο τρόπο υλοποιείται.
- **SSIS:** Παρουσιάζεται το SSIS μέσα στο Visual Studio καθώς μέσα από αυτό επιτυγχάνεται η ορθή υλοποίηση του ETL.
- **OLAP:** Παρουσιάζεται σε γενικές γραμμές τι είναι η τεχνολογία OLAP και τι περιλαμβάνει.
- **OLAP Cube SSAS:** Σε αυτό το σημείο δίνουμε μεγάλη έμφαση με εκτενή tutorials για την υλοποίηση ενός OLAP Cube μέσα στο περιβάλλον του Visual Studio αναλύοντας όλες τις δυνατότητες που μας προσφέρει αυτή η πλατφόρμα.
- **OLAP in Excel:** Τέλος παρουσιάζονται κάποιες τεχνικές για το πώς μπορούμε να χρησιμοποιήσουμε OLAP μέσα στο Microsoft Excel.

## ΕΠΙΛΟΓΟΣ

Στο κεφάλαιο αυτό είδαμε τεχνικά παραδείγματα χρησιμοποιώντας τις εφαρμογές που προσφέρει η Microsoft για να δημιουργήσουμε την βάση δεδομένων με το SQL Management Studio 2014, την αποθήκη δεδομένων με το SQL Management Studio 2014, το ETL με το Visual Studio 2013 και SSIS και τέλος τον κύβο με το Visual Studio 2013 και SSAS με την χρήση του παραδείγματος Hotel Management και του παραδείγματος Adventure Works.

## ΣΥΜΠΕΡΑΣΜΑΤΑ

Οι αποθήκες δεδομένων και τα συστήματα υποστήριξης αποφάσεων από την στιγμή που δημιουργήθηκαν μέχρι σήμερα κερδίζουν συνεχώς έδαφος. Υπάρχει πλέον η αναγκαιότητα από τους κρατικούς και ιδιωτικούς οργανισμούς για αξιοποίηση των δεδομένων που στοιβάζονται στις βάσεις δεδομένων για την κατανόηση προβλημάτων, μοτίβων, συμπεριφορών και την λήψη αποφάσεων για μελλοντικές στρατηγικές.

Σε αυτή την πτυχιακή μελετήσαμε σε βάθος την τεχνολογία OLAP με το πακέτο εργαλείων της Microsoft που προσφέρουν ένα ικανοποιητικό και ολοκληρωμένο σύνολο εφαρμογών για ένα σύστημα υποστήριξης αποφάσεων. Είδαμε πως περάσαμε από τις σχεσιακές βάσεις δεδομένων που χρησιμοποιούνταν για reports, στις αποθήκες δεδομένων για την τεχνολογία OLAP. Αναλύσαμε τους τρόπους που αυτό βελτιώνει τις λειτουργίες της επιχείρησης βελτιώνοντας την απόδοση των συστημάτων. Εξετάσαμε την μεθοδολογία του ETL κατά την διαδικασία μεταφοράς των δεδομένων. Τέλος δημιουργήσαμε τον OLAP κύβο.

Όλα αυτά αποτελούν αναπόσπαστα κομμάτια της τεχνολογίας OLAP που δημιουργήθηκαν για να υποστηρίξουν τα συστήματα λήψης αποφάσεων μιας επιχείρησης επιτυγχάνοντας γρήγορα και αποδοτικά reports.

## ΒΙΒΛΙΟΓΡΑΦΙΑ

Eric Thomsen (2002), John Wiley & Sons, OLAP Solutions: Building Multidimensional Information Systems, Edition 2, illustrated.

Robert Wrembel, Christian Koncilia, Idea Group Inc (IGI) 2007, Data warehouses and OLAP: Concepts, Architectures, and Solutions

[https://en.wikipedia.org/wiki/Online\\_analytical\\_processing](https://en.wikipedia.org/wiki/Online_analytical_processing)

Accessed 05/03/2016

[https://en.wikipedia.org/wiki/OLAP\\_cube](https://en.wikipedia.org/wiki/OLAP_cube)

Accessed 05/03/2016

[https://en.wikipedia.org/wiki/Microsoft\\_Analysis\\_Services](https://en.wikipedia.org/wiki/Microsoft_Analysis_Services)

Accessed 05/03/2016

[https://en.wikipedia.org/wiki/Data\\_warehouse](https://en.wikipedia.org/wiki/Data_warehouse)

Accessed 09/03/2016

<http://research.microsoft.com/pubs/76058/sigrecord.pdf>

Accessed 09/03/2016

[https://technet.microsoft.com/en-us/library/hh916543\(v=sc.12\).aspx](https://technet.microsoft.com/en-us/library/hh916543(v=sc.12).aspx)

Accessed 09/03/2016

[https://technet.microsoft.com/en-us/library/hh495576\(v=sc.12\).aspx](https://technet.microsoft.com/en-us/library/hh495576(v=sc.12).aspx)

Accessed 10/03/2016

<https://www.accelebrate.com/library/tutorials/ssas-2008>

Accessed 10/03/2016

<http://www.netwoven.com/2014/06/how-to-prepare-a-simple-olap-cube-using-sql-server-analysis-services/>

Accessed 10/03/2016

<http://www.it.uu.se/research/group/udbl/Theses/SobhanBadiozamyBSc.pdf>

Accessed 10/03/2016



<http://digilib.teiemt.gr/jspui/bitstream/123456789/2841/1/022006x03x156.pdf>

Accessed 20/03/2016

[https://dspace.lib.uom.gr/bitstream/2159/13821/1/losifidis\\_Msc2010.pdf](https://dspace.lib.uom.gr/bitstream/2159/13821/1/losifidis_Msc2010.pdf)

Accessed 20/03/2016

<http://www.wikihow.com/Create-OLAP-Cube-in-Analysis-Services>

Accessed 20/03/2016

[http://repfiles.kallipos.gr/html\\_books/72/kef11.html](http://repfiles.kallipos.gr/html_books/72/kef11.html)

Accessed 21/03/2016

[http://index.lib.teithe.gr:8080/bitstream/handle/10184/10668/Daravigkas\\_Dimitris.pdf?sequence=1](http://index.lib.teithe.gr:8080/bitstream/handle/10184/10668/Daravigkas_Dimitris.pdf?sequence=1)

Accessed 21/03/2016

<http://www.slideshare.net/TalitaLima1/ebook-pdf-olap>

Accessed 21/03/2016

[https://books.google.com.cy/books?id=XFivorxZDm8C&source=gbs\\_navlinks\\_s](https://books.google.com.cy/books?id=XFivorxZDm8C&source=gbs_navlinks_s)

Accessed 21/03/2016

[https://docs.oracle.com/cd/B10500\\_01/server.920/a96520/concept.htm](https://docs.oracle.com/cd/B10500_01/server.920/a96520/concept.htm)

Accessed 10/03/2016

## ΟΔΗΓΟΣ ΧΡΗΣΗΣ ΛΟΓΙΣΜΙΚΟΥ

Η πτυχιακή αυτή υλοποιήθηκε σε μία εικονική μηχανή στην οποία εγκαταστάθηκαν το σύνολο των λογισμικών που απαιτούνται. Περιληπτικά τα λογισμικά είναι:

- Virtual Box
- Windows 7 professional N x86
- Microsoft Office 2007 Enterprise x86 Service Pack 1
- Excel plug-in OLAP PivotTable Extensions
- SQL Server 2014 standard edition x86 Service Pack 1
- SQL Server Data Tools
- Visual Studio 2013 professional
- Silverlight
- Adobe Reader PDF
- Google Chrome
- Media Feature Pack for Windows 7 N Service Pack 1

Στην πτυχιακή αυτή χρησιμοποιήθηκε το Virtual Box για την δημιουργία ενός virtual απομονωμένου περιβάλλοντος. Στο VB εγκαταστάθηκαν windows 7 professional με service pack 1. Τα λογισμικά όλα κατέβηκαν από το Dream Spark / MSDNAA με τους κωδικούς της σχολής. Τα ISO αρχεία των εφαρμογών κατέβηκαν από το Dream Spark στο Host μηχάνημα του Virtual Box. Στο Virtual Box στο μενού devices -> optical drives επιλέγουμε το κάθε ISO αρχείο των εφαρμογών να γίνει mount για να μπορούμε να δούμε το setup μέσα στο guest μηχάνημα (windows 7).

Για server της βάσης δεδομένων εγκαταστάθηκε SQL Server 2014 standard edition με Service Pack 1 x86. Για το development εγκαταστάθηκε Visual Studio 2013 professional μαζί με Business Intelligence tools (SQL Server Data Tools). Για την σωστή λειτουργία είναι απαραίτητη η εγκατάσταση του Microsoft Silverlight. Επίσης εγκαταστάθηκε Microsoft Office Enterprise 2007.

Η διαχείριση της βάσης και της Αποθήκης δεδομένων έγινε με το Microsoft SQL Management Studio 2014. Για το ETL χρησιμοποιήθηκε Microsoft Visual Studio και integration services (SSIS).

Η ανάπτυξη του κύβου και η διαχείριση του έγινε με Visual Studio 2013 και Analysis Services (SSAS). Η Ανάλυση του κύβου επίσης έγινε και στο Microsoft Office Excel 2007.

Επιπλέον υλικό που χρησιμοποιήσαμε για εκπαιδευτικούς λόγους σε αυτή τη πτυχιακή είναι η βάση δεδομένων Adventure Works 2014 της Microsoft, η αποθήκη δεδομένων Adventure Works DW 2014 της Microsoft και το project του Visual Studio Adventure Works DW 2014 Multidimensional-SE.

Ο κωδικός για τα windows 7 στο virtual machine είναι "olap".

Για τα credentials για την σύνδεση στο Management Studio χρησιμοποιούμε windows authentication.

Στα project στο Visual Studio όπου ζητήθηκε username και password ορίσαμε μόνο το username : olap και το password το αφήσαμε κενό.