



# ניתוח מערכת - מדרישות למודלים

חלק א'

# מנתח מערכות ותוצרי הגדרת דרישות

- מנתח מצב קיים ובעיות הדורשות מחשוב
- מגדיר איפיון למרחב הבעיה וקווי מתאר לפתרון
- תוחם ומחדד מטרות מערכת
- אוסף ומסווג דרישות מלקוחות ומשתמשים

## • תוצרים פורמליים:

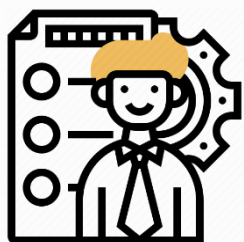
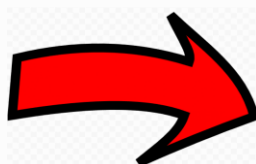
• מסמך ייזום ✓

• ניתוח מצב קיים

• הגדרת דרישות ✓

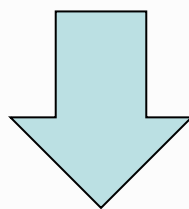
• מודלים, דיאגרמות ופורמטים סטנדרטיים

• איפיון פתרון



## שלב הניתוח

- תהליכי המחשב שתבצע המערכת (הפונקציות)
- הקלטים של המערכת והמקורות שלהם
- הפלטים של המערכת והיעדים שלהם
- הנתונים שייאגרו בבסיסי הנתונים של המערכת



להגדיר במדויק את ביצועי המערכת

# תרגום דרישות למודלים - גישות

## גישת התהליכים

- פירוק המערכת לתהליכים ופונקציות
- דגש על זרימה וטרנספורמציה של נתונים

## גישת ישויות הנתונים

- פירוק המערכת לחבילות נתונים המייצגות ישויות אבסטרקטיות או ממשיות
- דגש על מבנה חבילות הנתונים ובקרת ביצוע

## גישות משולבות

- המערכת מיוצגת על ידי ישויות, הגדרת יחסים מבניים בין הישויות, כל יישות משתתפת בתהליכים המשנים את נתוניה

# גישה תהליכית – Data Flow

## גישת Data Flow Modeling

- תיאור התנהגותי של המערכת מנקודת מבט זרימת הנתונים, מהישויות החיצוניות אל פונקציות המבצעות עליהם טרנספורמציות ועד לאחסון => תרשים DFD
- תיאור כל הישויות החיצוניות באינטראקציה עם המערכת => תרשים קונטקסט SCD
- מילונים מלווים את הפירוק
- מילון פונקציות, מילון מאגרי מידע, מילון ישויות

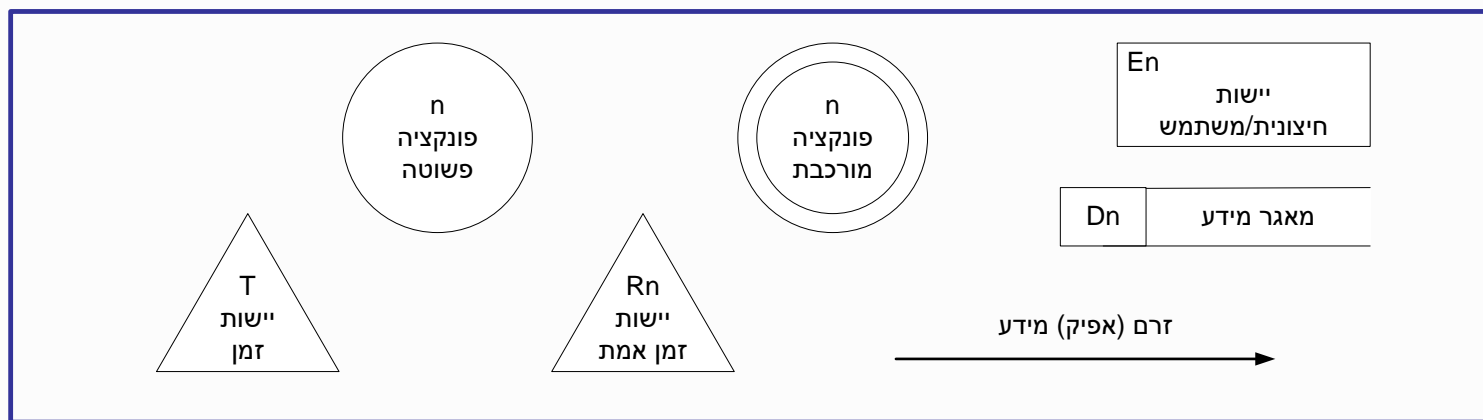
## מתודולוגיות הדוגלות ב- Structured Analysis (SSA) מתבססות על גישה זו

כיום השימוש בתרשימי DFD ו- SCD אינו נפוץ במחוזותינו

# מרכיבי תרשים DFD

תרשים DFD מורכב מחמישה סמלים עיקריים:

- עיגול (יחיד או כפול) מציין פונקציה (פשוטה או מורכבת) שהמערכת מבצעת
- מלבן מציין ישות חיצונית או משתמש במערכת
- מלבן מוארך ובו משבצת בצד אחד ופתח בצד האחר מציין מאגר מידע
- משולש מציין יחידת זמן או מכשיר הקשור למערכת הפועל בזמן אמת
- חץ מציין זרם מידע (אפיק מידע) בין פונקציות למרכיבים אחרים במערכת



# דוגמא: מערכת מידע לוועד בית

יש להקים מערכת מידע לוועד בית משותף. המערכת תטפל בתשלומי הדיירים.

המערכת תאפשר הזנת פרטי דיירי הדירות בבניין וכן עדכון פרטי הדיירים, כולל הוספות, שינויים וביטולים. על פי פרטי הדיירים בדירה יקבע נציג ועד הבית תעריף התשלום החודשי הנדרש מהדייר ואת מועד תחילת התשלום ויזין אותם למערכת.

דיירי הבניין מעבירים תשלומים לחברי הוועד בהמחאה או במזומן. המערכת תאפשר לחבר הוועד להזין את פרטי התשלום (כולל הדייר המשלם, סכום ותאריך) ותפיק קבלה לדייר.

המערכת תאפשר להפיק, לפי דרישה, דוחות תשלומים לתקופות שונות, לכל הדיירים או לחלק מהם. היא תאפשר לחבר הוועד להזין תקופת דוח וזיהוי דיירים. לכל דייר היא תפיק פרטי התשלומים שהתקבלו ממנו במשך התקופה ויתרת החוב לסופה.

בסוף כל חודש תפיק המערכת דוח יתרות חוב של כל הדיירים. לכל דייר שיתרת החוב שלו גבוהה מסכום של שלושה חודשי תשלום תופק תזכורת עם יתרת החוב. כן תופק תזכורת לדייר שיתרת חובו גבוהה מסכום של שני חודשי תשלום אם הוא קיבל תזכורת גם בחודש הקודם.








# גישה תהליכית מודרנית – Use Cases

הגישה החדשה למידול: Use Case Model 

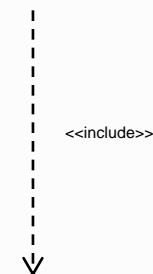
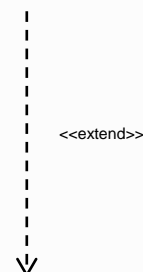
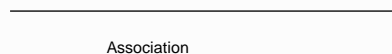
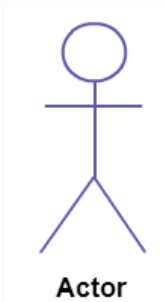
תאור התנהגותי של מהלך שימוש במערכת ע"י  
משתמש (שחקן \ תרחיש) או תיאור מהלך עסקי שלם

תיאור הכולל הגדרת מטרה, משתמש, תנאים מקדימים  
לתרחיש, טריגר לתרחיש, מהלך עניינים אפשרי ומה  
יהיה מצב המערכת בסיומו. 

([https://www.methoda.cloud/methoda.cloud/originals/1559252959H\\_UML\\_Sdoc02.pdf](https://www.methoda.cloud/methoda.cloud/originals/1559252959H_UML_Sdoc02.pdf))

תרשימים סטנדרטיים באמצעות UML 

# מרכיבי תרשים Use Case



## שחקנים

- ישות חיצונית אשר יש לה קשר עם המערכת
- לכל ישות יש תפקיד ביחס למערכת
- השחקנים מייצגים את תיחום המערכת אך הם לא חלק ממנה
- שחקן יכול להיות ישות חיצונית, מערכת אחרת או כל אמצעי מחשוב-
  - עובד, לקוח, אורח
  - ארגון- מדור בחברה, חברת משלוחים
  - התקן- קופה רושמת, מדפסת
  - מערכת חיצונית- מערכת כספים, מלאי, מערכת נוכחות עובדים

## איך נמצא את השחקנים

- מי משתמש במערכת?
- מי מתקין את המערכת?
- מי מפעיל את המערכת?
- מי מתחזק את המערכת?
- מי מכבה את המערכת?
- באילו מערכות אחרות משתמשים במערכת זו?
- מי מקבל מידע ממערכת זו?
- מי מספק מידע למערכת?
- האם קורה משהו באופן אוטומטי ברגע הנוכחי?

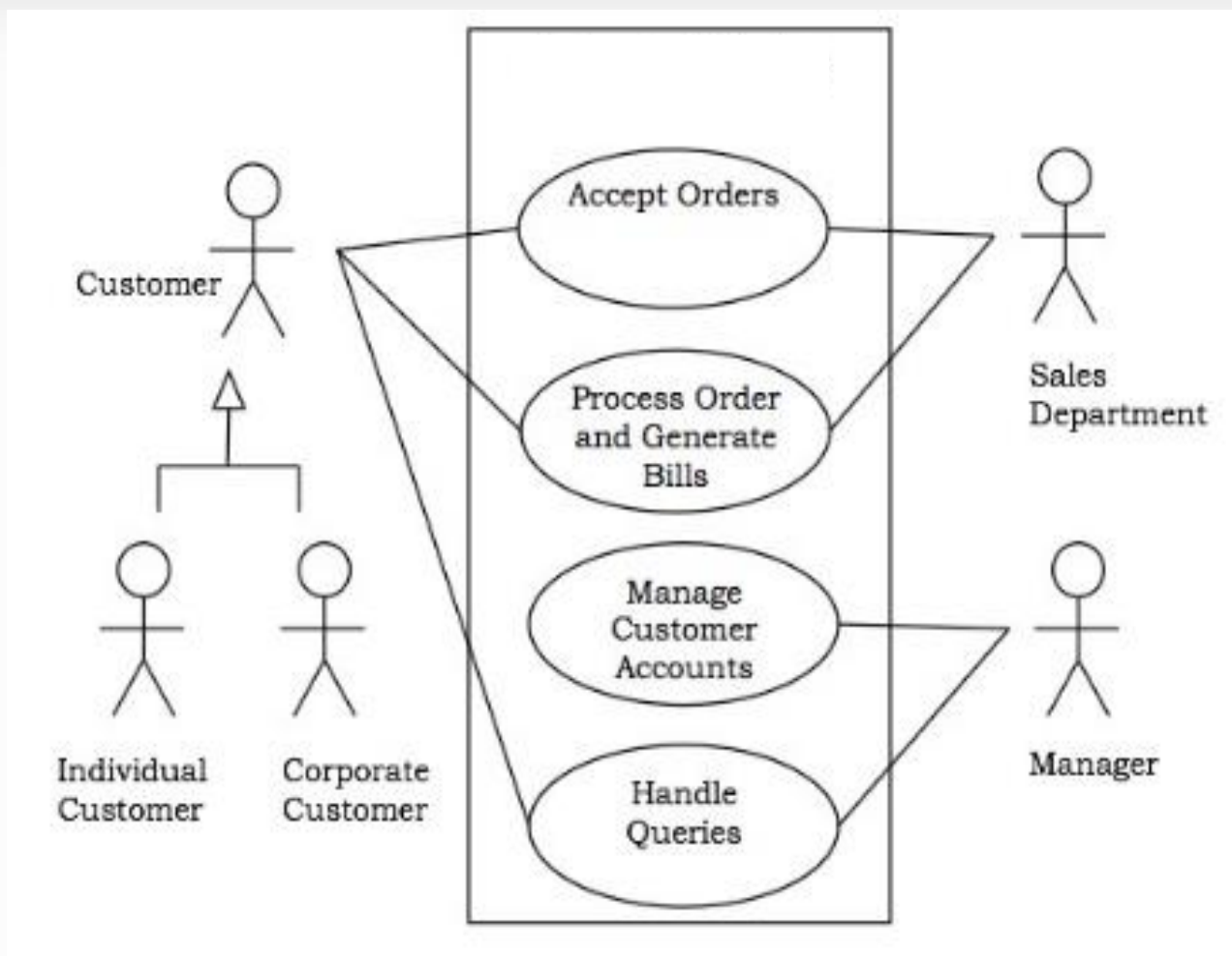
## דוגמאות ל- use cases

- משיכת מזומן מכספומט (השחקן- לקוח של הבנק)
- הרשמה בכניסה למלון (השחקן- אורח)
- הפקת דוח מכירות (השחקן- מנהל המכירות)
- קניית מצרכים בסופרמרקט (השחקן- לקוח של הסופר)
- שליחת דרישת תשלום (השחקן- הספק)
- קביעת מסגרת אשראי ללקוח (השחקן- מחלקת אשראי)
- דיווח תקלות במבנה (השחקן- מהנדס/ מפקח בנייה)

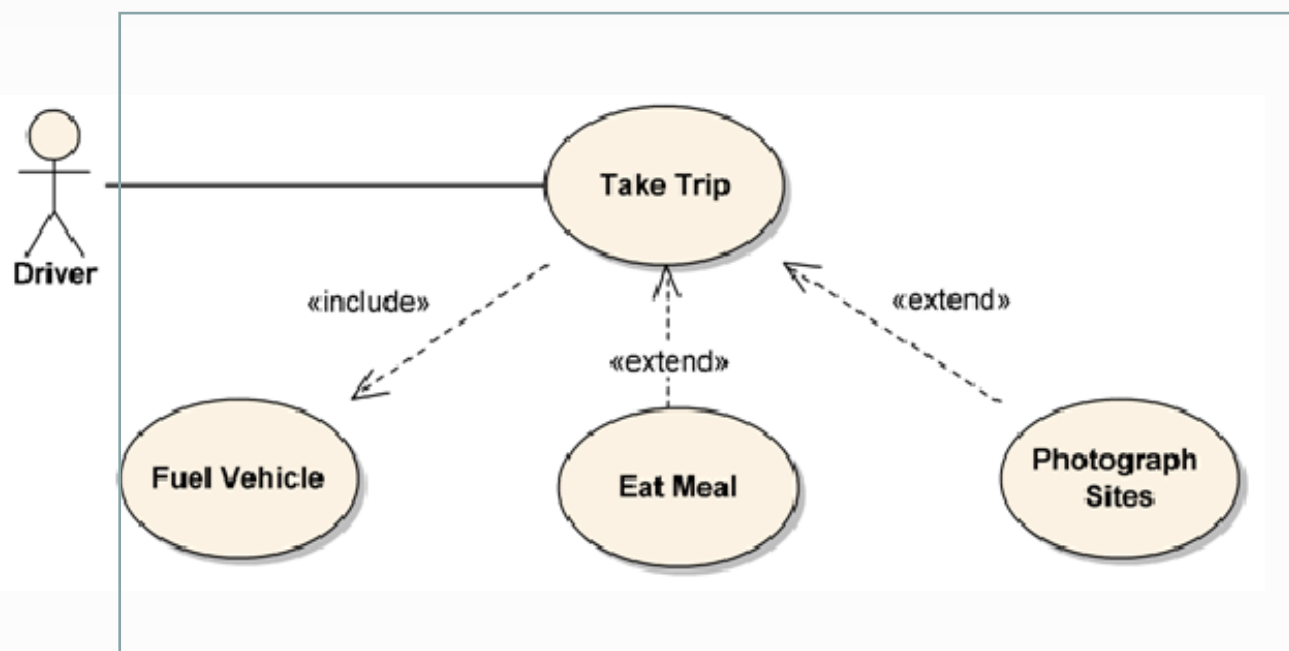
## מערכת הזמנות

- מערכת ממוחשבת של הזמנות, ניהול מלאי ולקוחות:
- הלקוחות יכולים להיות פרטיים או עסקיים
  - הלקוח עושה הזמנה, מחלקת מחירות מבצעת אותה ויוצרת חשבון- הלקוח מקבל את החשבון
  - המנהל מנהל את חשבונות הלקוחות ועונה על שאלות ששואלים הלקוחות.

# דוגמא

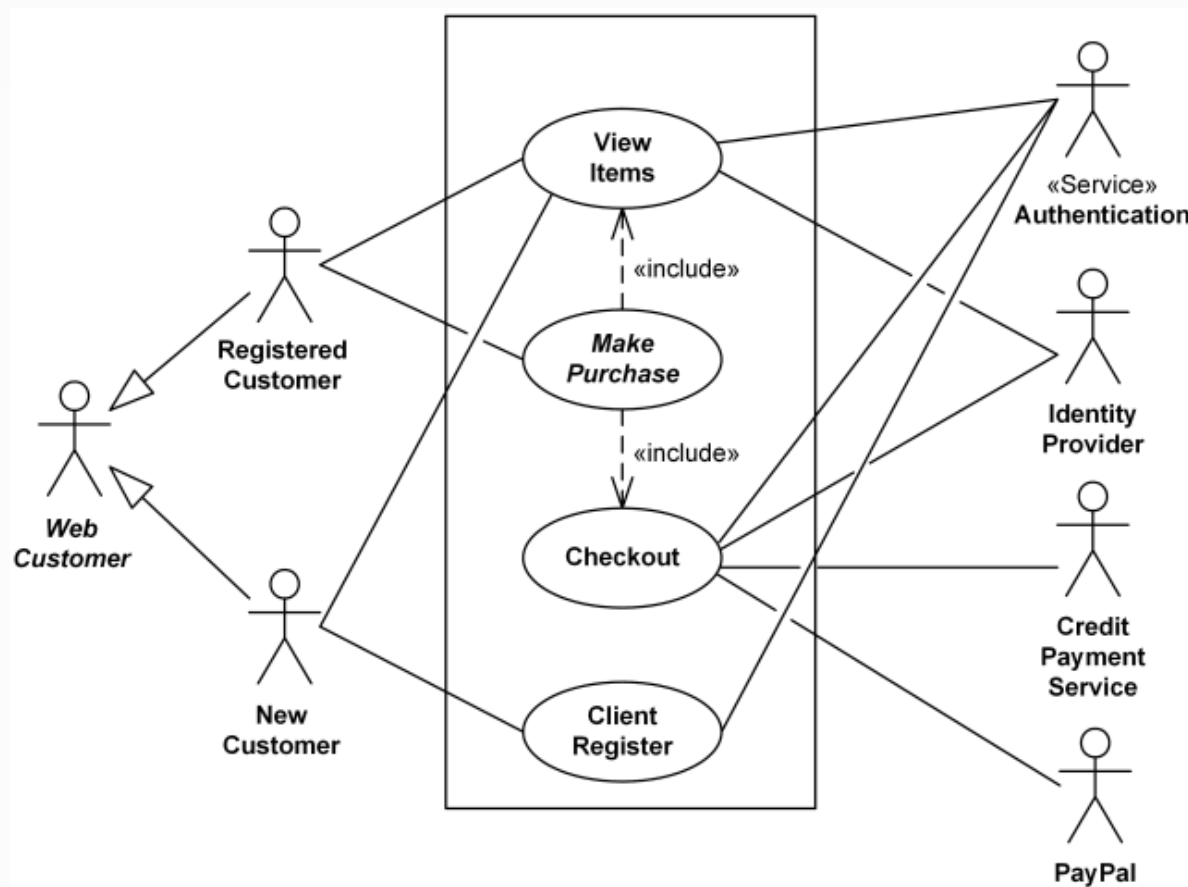


# <<extend>> & <<Include>>





# דוגמא נוספת




# גישה מבוססת נתונים Entity Relations Model

## Objects & Data Stores

• תיאור המערכת כאוסף מאגרי מידע שאפשר להפעיל  
בכדי לבצע עיבודים

• פירוק המערכת לאובייקטים המעבירים מסרים זה לזה  
במהלך העיבוד ומבקשים שירותים זה מזה

• שימוש בתרשימי ERD –Entity Relations Diagrams 

• שימוש מודרני בתרשימים סטנדרטים במסגרת UML  
כגון Activity Diagrams – Class Diagrams 

• מתודולוגיות הדוגלות ב- Object Oriented Analysis  
(OOA) מתבססות על גישה זו 

# סיווג נתונים

🌐 ללא מבנה: מסמכים בעלי טקסט חופשי, דפי HTML , או תמונות וסרטים.

🌐 מובנים: הגדרה ברורה של מבנה (סכימה) , בפרט נתונים במסד נתונים רלציוני.

🌐 מובנים למחצה : Mail's , דפי XML הכוללים הגדרות המשרות הררכיה או משמעות (Meta Data)

# פרספקטיבות וסוגי מודלים

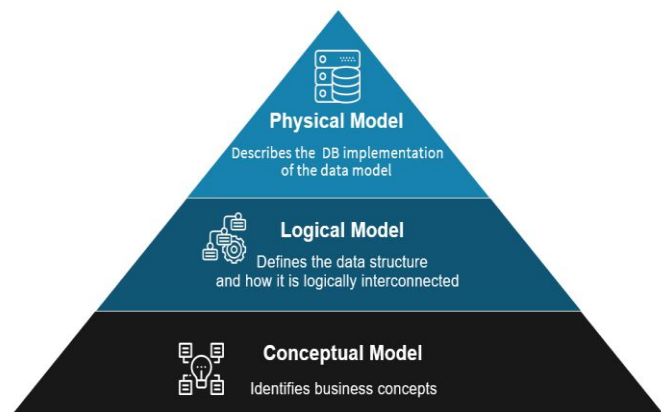
## שלוש רמות מודלים:

### מודל קונספטואלי:

- תיאור סמנטי "טהור" של תחום הבעיה (משמעות)
- תיאור יישויות (Entities), תכונות (Attributes) וקשרים (Relationships)

### מודל לוגי:

- ייצוג תחום הבעיה במושגי טכנולוגיית
- ניהול הנתונים (לדוגמא: טבלאות, רשומות)



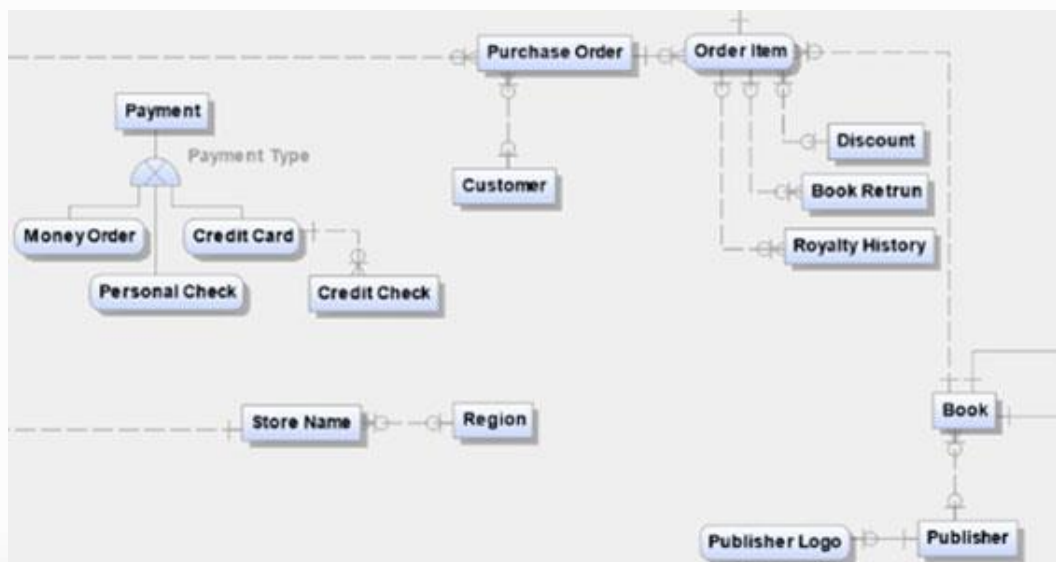
### מודל פיסי:

- כיצד הנתונים נשמרים ב-HD או
- במסגרת מאפייני ניהול RDBMS מסחרי ספציפי

# מודל קונספטואלי לנתונים

תאור מה מכילה המערכת ברמת מושגים, ישויות והיחסים ביניהם.

מספקת פרספקטיבה על מבנה הארגון באמצעות הגדרת יחסים בין הישויות העסקיות הקיימות או מתוכננות



# מודל לוגי

• תאור נתונים במושגי הטכנולוגיה וגישת ניהול הנתונים.

• הבולט מבין שלושת סוגי המודלים, יתורגם פעמים רבות למודל רלאציוני עקב תפוצת מסדי נתונים מבוססי SQL או למודלים מסוג NoSQL.

• חסרון מרכזי: אובדן חלק נכבד ממרכיבי המשמעות במסגרת תהליך הפיתוח יחסית למודל הקונספטואלי



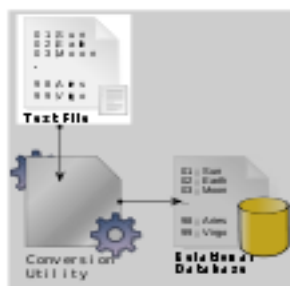
# גישות לייצוג (מודל) נתונים לוגי

שטוח (קבצים) 🌐

היררכי 🌐

רשתי 🌐

טבלאי 🌐



Flat model



Hierarchical model



Network model



Relational model

# מודל פיסי לנתונים

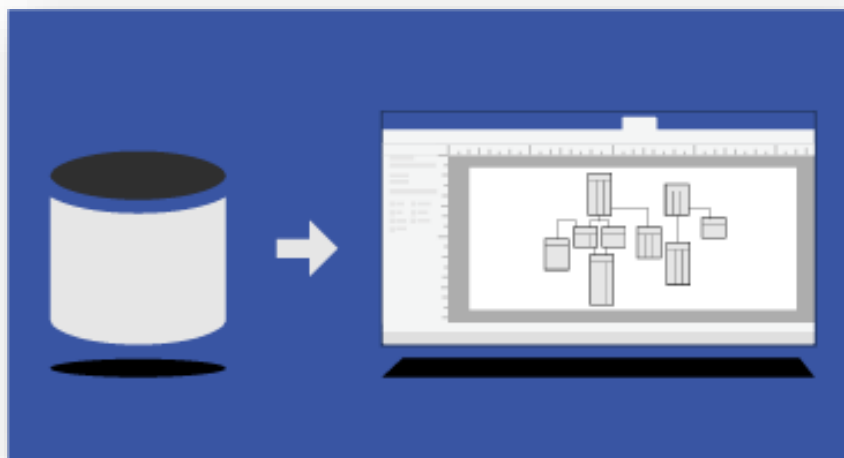
- תאור האספקטים הפיסיים לאחסון הנתונים
- הגדרת מקום וחוקי הקצאת מקום , חלוקה פיסית של אחסון טבלאות וכו'
- לרוב פחות משמעותי בשלבי הניתוח והעיבוד
- אופטימיזציה של מימוש ע"י מומחים בתחום באמצעות כלים המלווים RDBMS מסחריים





# סדר יצירה ומעבר בין המודלים

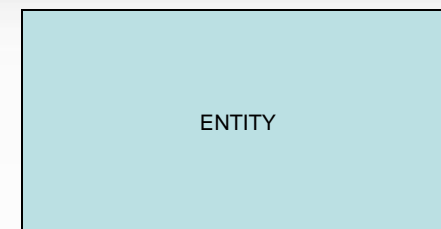
- פיתוח מודל קונספטואלי באמצעות E-R (Relations Entity).
- תרגום למודל לוגי לצורך מימוש , לרוב במסד נתונים ובכלל זה גם תהליך הנקרא נירמול
- ביצוע התאמות למודל הפיזי מתוך הגדרות המודל הלוגי



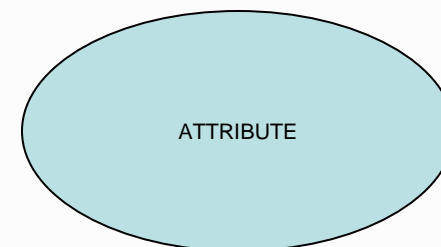
- תאור קונספטואלי של נתונים ויחסים
- גישת Top-Down
- משמש כבר בשלב איסוף הדרישות לתאור הנתונים שיידרשו או יאוחסנו במסד הנתונים.
- יכול לשמש לתאור כל אוסף של רעיונות ויחסים במסגרת תחום כלשהוא.
- יתורגם למודל לוגי כדוגמת המודל הרלציוני
- או יתורגם למודל אובייקטים

# מרכיבי E-R Diagram

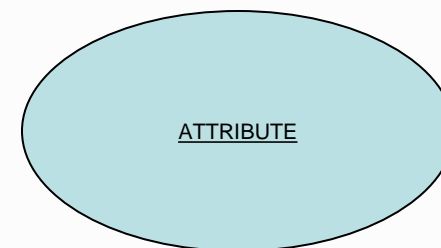
יישות ממשית או קונספטואלית עבורה נרצה לאחסן נתונים



תכונה המאפיינת יישות

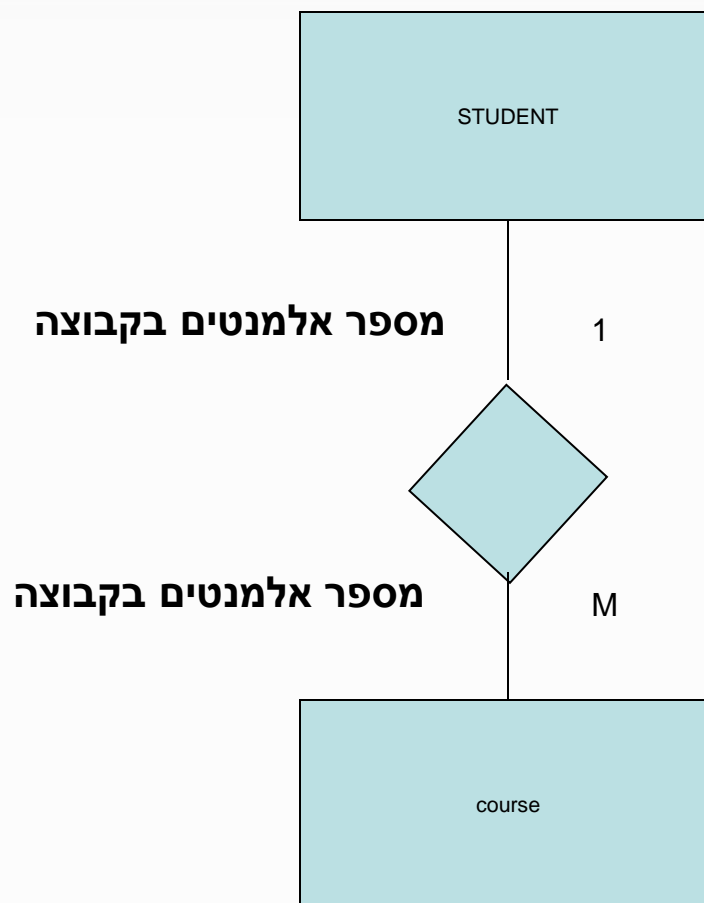


תכונת מפתח ייחודית היוצרת הבחנה בין ישויות שונות

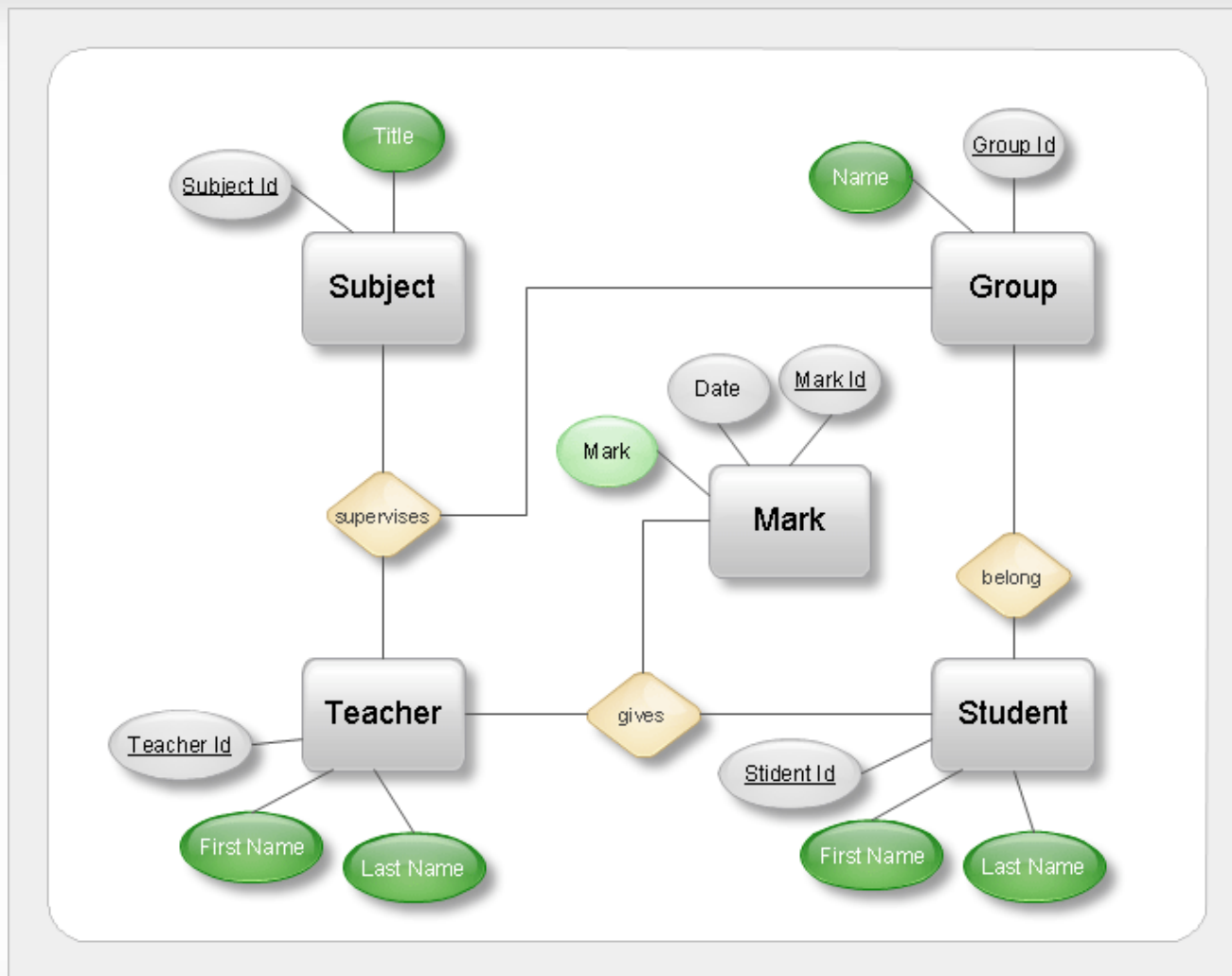


# יחסים E-R Diagram

יחס המתאר כיצד זוג ישויות  
משתפים אינפורמציה



# E-R Diagram



# גישות מעורבות - Object Process Methodology

• מידול תהליכים ומידול אובייקטים בכפיפה אחת

• כולל 2 סוגי אלמנטים : ישויות וקשרים

• סוגי ישויות :

• אובייקט - ישות המתקיימת במערכת לזמן נתון

• מצב – סיטואציה שבה יכול להמצא האובייקט

• תהליך – פעילות המשנה את מצבו של האובייקט

• סוגי קשרים :

• קשר מבני – חיבור קבוע בין אובייקטים

• קשר פרוצדורלי - חיבור אובייקט לתהליך לשינוי מצבו

• בסיס לגישת Model Based Systems Engineering

# גישות מעורבות - Object Oriented Analysis

- מיועד לניתוח דרישות פונקציונליות (מה):
- הדרישות מחוברות לאובייקטים באופן המשלב תהליכים (Processes) עם נתונים (State)
- בחינת תחום הבעיה (Domain), התמקדות בתחום הליבה הדורש התמחות וטיפול ספציפי
- יצירת מודל קונספטואלי לנתונים בתחום המנותח
- אנקפסולצית השירותים העסקיים במערכת
- הניתוח מתחיל ממסמך SRS או תהליך ישיר מול מרואיינים
- תוצרי הניתוח מתארים פונקציונליות המערכת וכוללים:
  - דיאגרמות UC
  - דיאגרמת מחלקות ( Class Diagrams )
  - דיאגרמות אינטראקציה כדוגמת Sequence Diagrams
  - תאור מסכים



OBJECT-ORIENTED

# דוגמא



- יישות טיסה
- מאפיינים:
  - קו טיסה
  - תאריך טיסה
  - מוצא
  - יעד
  - סטטוס
  - תפוסה



# דוגמא



- יישות טיסה

- פעילויות:

- פתח

- בטל

- שנה יעד

- שנה תאריך טיסה

- עדכן סטטוס

## דוגמא



- יישות טיסה + מצב

- מאפיינים:

- קו טיסה 232

- תאריך טיסה 01/01/2022

- מוצא תל אביב

- יעד ונציה

- סטטוס פתוחה

- תפוסה 322

## דוגמא



- יישות טיסה + מצב

- מאפיינים:

- קו טיסה 232

- תאריך טיסה 01/01/2022

- מוצא תל אביב

- יעד ונציה

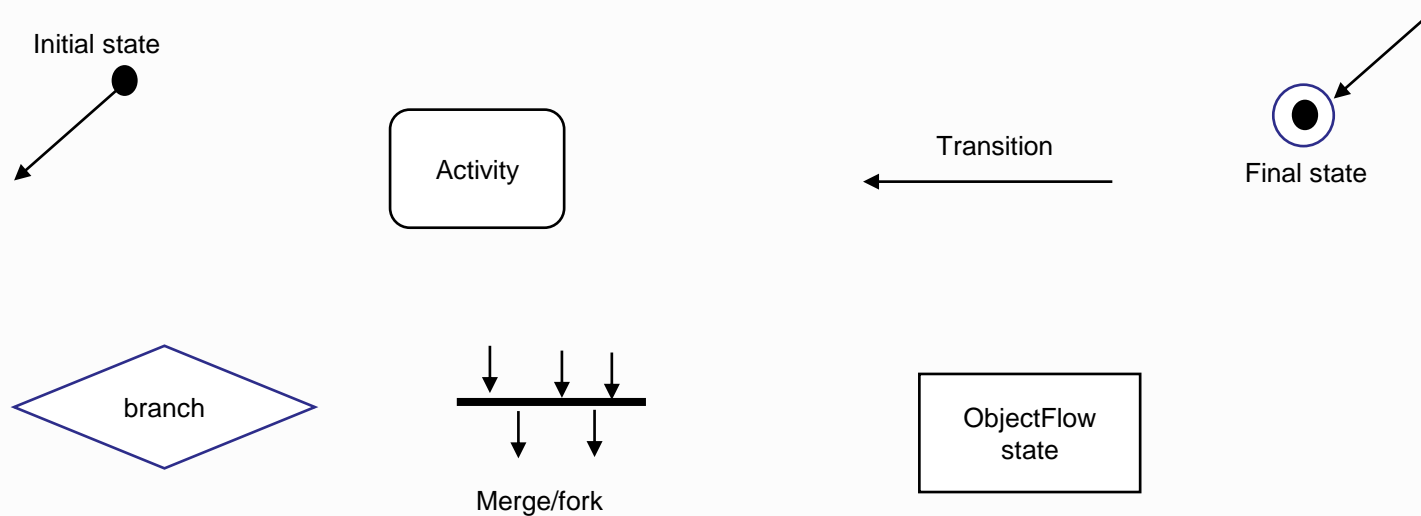
- סטטוס סגורה

- תפוסה 400

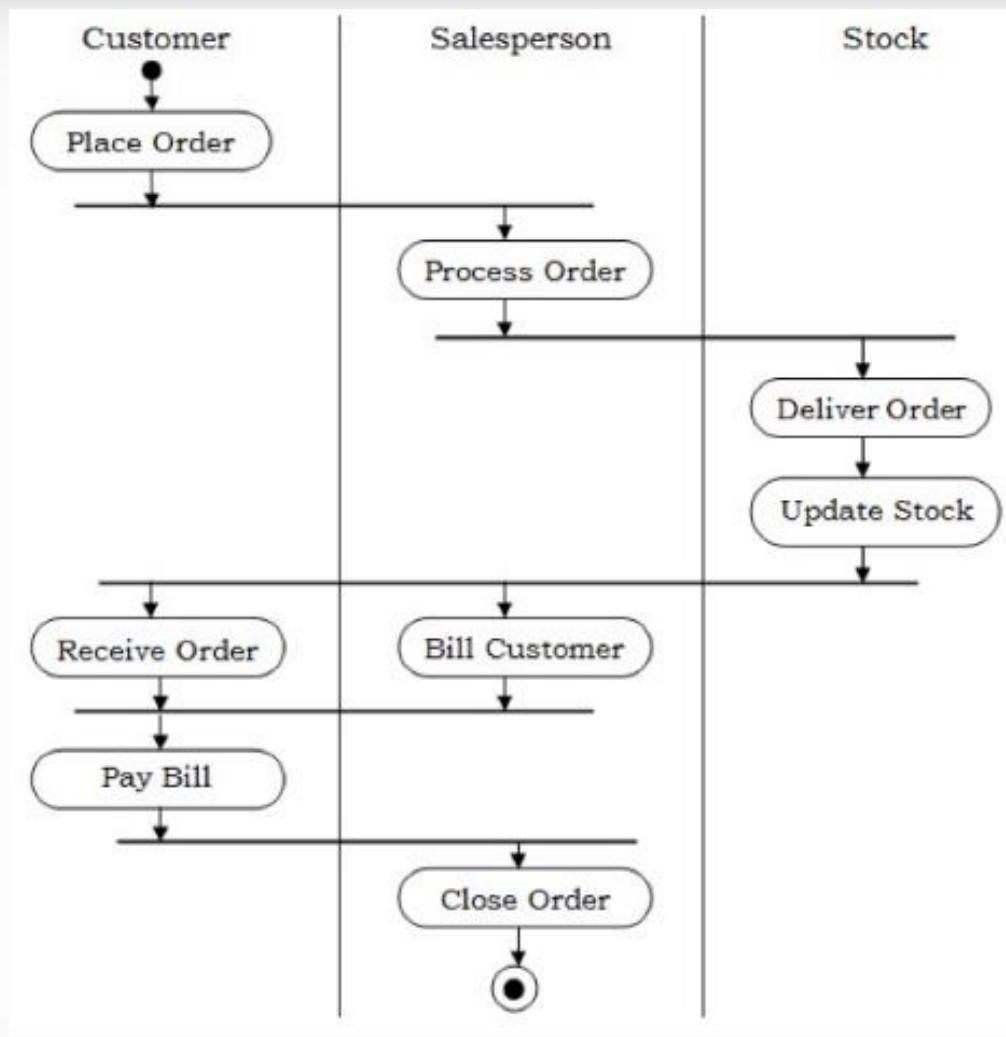
# Activity Diagram

- תרשים פעילות מציג את הזרימה מפעילות לפעילות בתחום המערכת ובין השחקנים
- דרך להצגת פעילויות במערכת בדומה ל-work flow
- כל פעילות ב-use case הופכת לתרשים פעילות אחד

# שיטת הסימון



# Activity Diagram



# State Machine Diagram

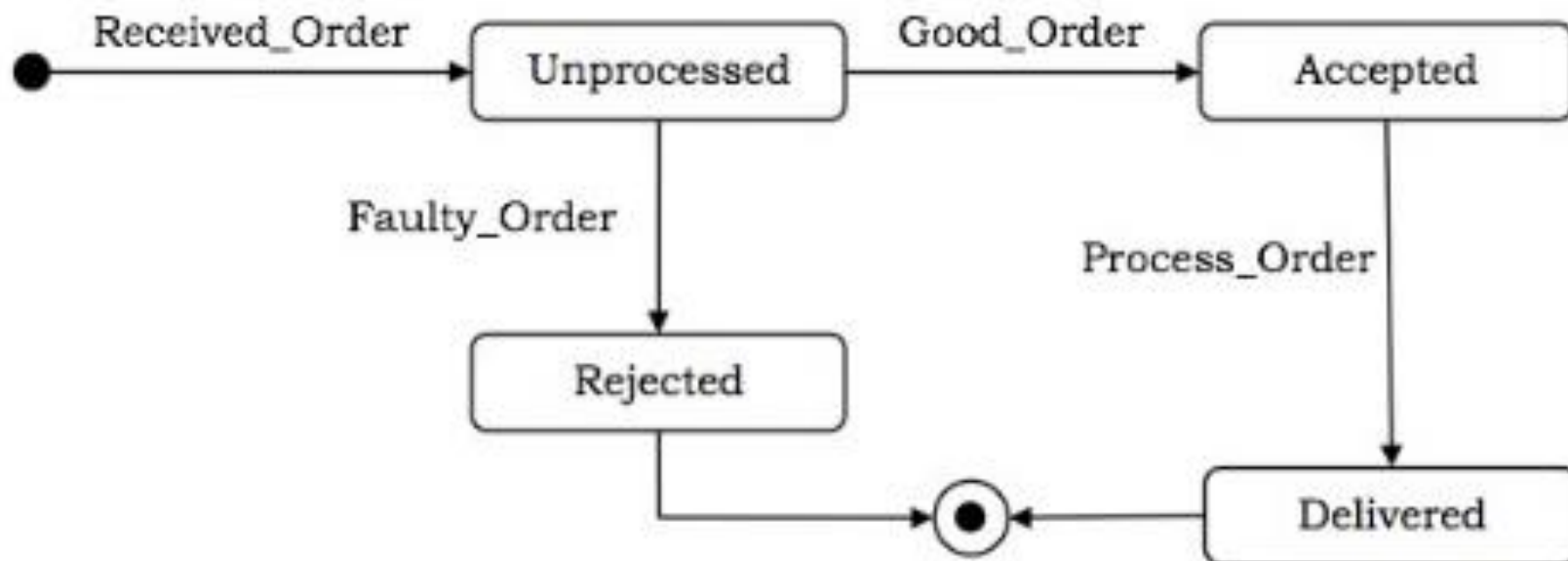
- בתרשים מצבים שמים את האובייקט במרכז
- התרשים מתאר את זרימת האובייקט ממצב אחד למשנהו ואת המצבים השונים שבו האובייקט יכול להיות לאורך חיי המערכת.
- תרשים זה הוא מבוסס על האוטומט של פרופסור דוד הראל
- כל תרשים מתאר מחלקה אחת

# שיטת הסימון





# State Machine Diagram



## סיכום ביניים

- התחלנו את שלב הניתוח בתהליך פיתוח תוכנה
- 3 גישות ניתוח
- דיאגרמות:
  - DFD (לא בשימוש כיום)
  - Use Case Diagram
  - ERD (מודל סוג 1)
  - Activity Diagram
  - State Machine Diagram

## בשבוע הבא..

- המשך דיאגרמות לניתוח המערכת