MONASH University

FIT1013 Digital Futures: IT for Business
Week 9 : Modularisation, Structure Charts,
Connecting to External Data

On completion of your study this week, you should aim to:
• Discuss program design approaches
• Design modules using structure charts
• Import data from text files

GROUP OF EIGHT AUSTRALIA Sections © 2016 Cengage Learning All rights reserved, © Thomson 2003

Two Approaches to System Development

MONASH University

▪ Traditional Approach
  – Also called structured system development
  – Structured analysis and design technique (SADT)
▪ Structured programming
  – Improves computer program quality
  – Allows other programmers to easily read and modify code
  – Each program module has one beginning and one ending

More details about Systems Development will be covered in **FIT2001**

2

Structured Analysis

MONASH University

▪ Define what system needs to do (processing requirements)
▪ Define data system needs to store and use (data requirements)
▪ Define inputs and outputs
▪ Define how functions work together to accomplish tasks
▪ Data flow diagrams (DFD) and entity relationship diagrams (ERD) show results of structured analysis

3

Structured Design

MONASH University

▪ Technique developed to provide design guidelines
  – What set of programs should be
  – What program should accomplish
  – How programs should be organized into a hierarchy
▪ Modules are shown with structure chart
▪ Main principle of program modules
  – Loosely coupled – module is independent of other modules
 – Highly cohesive – module has one clear task

4

The Structure Chart

- Describes functions and sub functions of each part of system
- Shows relationships between modules of a computer program
- Simple and direct organization
  - Each module performs a specific function
  - Each layer in a program performs specific activities
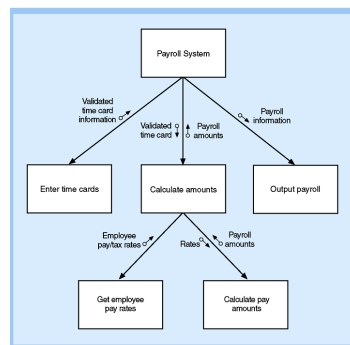- Chart is tree-like with root module and branches
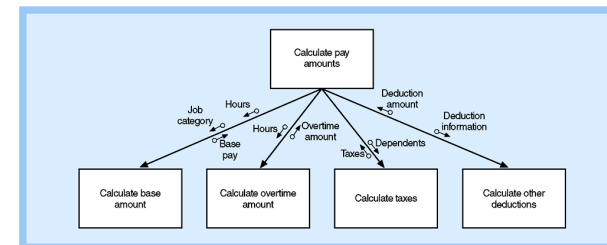
Structure Chart Symbols
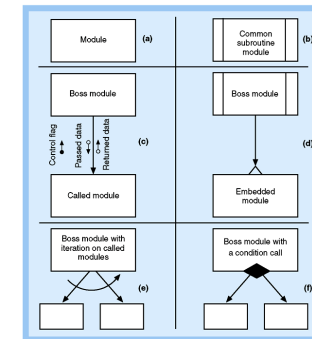
Structure Chart Created Using Structured Design Technique

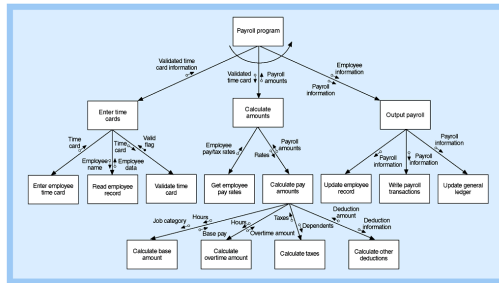**Example of a structure chart**

A Simple Structure Chart for the Calculate Pay Amounts Module

Structure Chart for Entire Payroll Program

Data Flow Diagram (DFD) created using Structured Analysis Technique
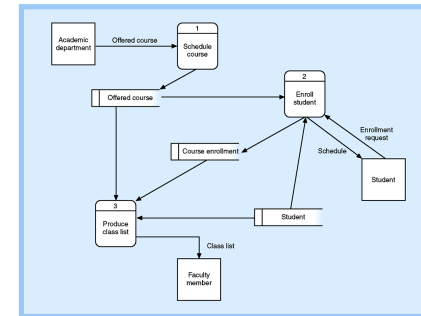
Note: **FIT2090** will cover the topic on DFD

Entity-Relationship Diagram (ERD) created using the Structured Analysis technique
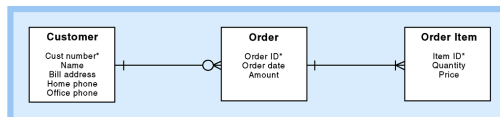
Note: ERD is covered in the **database units**

Object-Oriented Approach

- Views information system as collection of interacting objects that work together to accomplish tasks
  - Objects - things in computer system that can respond to messages
  - No processes, programs, data entities, or files are defined – just objects
- Object-oriented analysis (OOA)
  - Defines types of objects that do work of system
  - Shows how objects interact with users to complete tasks

Object-Oriented Approach to Systems

MONASH University

13

Object-Oriented Approach (continued)

MONASH University

- Object-oriented design (OOD)
  - Defines object types needed to communicate with people and devices in system
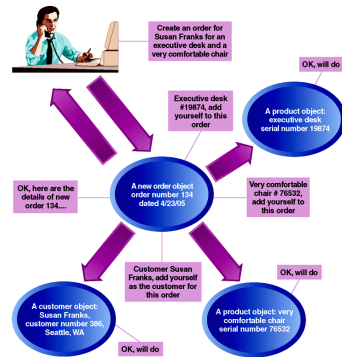 - Shows how objects interact to complete tasks
  - Refines each type of object for implementation with specific language of environment
- Object-oriented programming (OOP)
  - Writing statements in programming language to define what each type of object does
- Benefits of OOA include naturalness and reuse

14

Class Diagram Created During OO Analysis

MONASH University
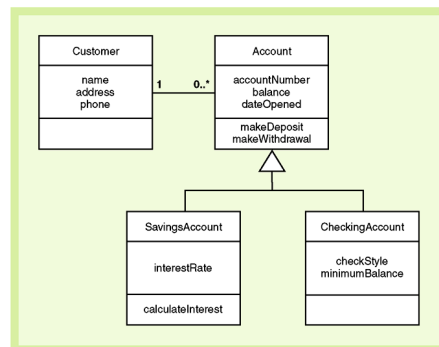
15

Developing a Structure Chart

MONASH University

- Transaction Analysis
  - Uses system flow chart and event table inputs
  - Upper-level modules developed first
  - Identifies each transaction supported by program
- Transform Analysis
  - Uses **DFD** fragments for inputs
  - Computer program 'transforms' inputs into outputs
  - Charts have input, calculate, and output subtrees

16

Steps to Create a Structure Chart from a DFD Fragment

MONASH University

- Determine primary information flow
  - Main stream of data transformed from some input form to output form
- Find process that represents most fundamental change from input to output
- Redraw **DFD** with inputs to left and outputs to right – central transform process goes in middle
- Generate first draft structure chart based on redrawn data flow

17

Data Flow Diagrams

MONASH University

- Data flow diagram: a graphical representation of a system that depicts the systems components; the data flows among the components; and the sources, destinations, and storage of data.
- Use a limited number of symbols.
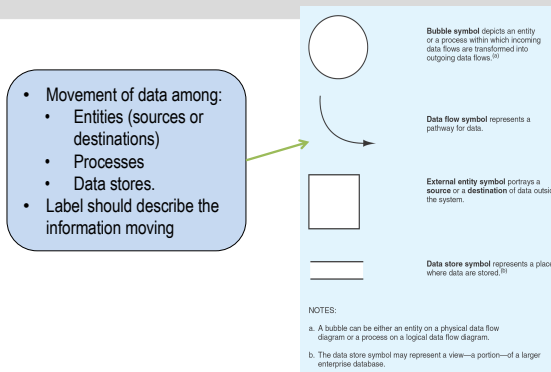- Do not depict management or operational elements of a system.

18

What are DFDs used for?

MONASH University

- A new (or part of an) implementation of a system
- A new (or part of a) **business process reengineering (BPR)** project
- As part of the assessment of internal controls (e.g. for auditing purposes)

Note: BPR will be discussed in **FIT2090**

Basic DFD Symbols

MONASH University

- Movement of data among:
  - Entities (sources or destinations)
  - Processes
  - Data stores.
- Label should describe the information moving

Bubble symbol depicts an entity or a process within which incoming data flows are transformed into outgoing data flows.[a]

Data flow symbol represents a pathway for data.

External entity symbol portrays a **source** or a **destination** of data outside the system.

Data store symbol represents a place where data are stored.[b]

NOTES:

a. A bubble can be either an entity on a physical data flow diagram or a process on a logical data flow diagram.

b. The data store symbol may represent a view—a portion—of a larger enterprise database.

20

Data Flow Diagram Levels

MONASH University

- Context
  - Highest level (most general)
  - Purpose: show inputs and outputs into system
  - Characteristics: one process symbol only, no data stores.
- Level-0
  - Purpose: show all major activity steps of a system
  - Characteristics: processes are labeled 1.0, 2.0 and so on.
- Level-1
  - Purpose: show one major activity divided into sub-activities
  - Characteristics: processes are labeled 1.1, 1.2 and so on.

More details are covered in FIT2090

21

Context Diagram

MONASH University

- Context diagram
  - top-level, and less detailed, diagram of a system depicting the system and all its activities as a single bubble and showing the data flows into and out of the system and into and out of the external entities.
- External entities
  - those entities (i.e., persons, places, or things) outside the system that send data to, or receive data from, the system.
- Internal entities
  - those entities within the system that transform data
  - Includes, for example, accounting clerks (persons), departments (places), and computers (things)
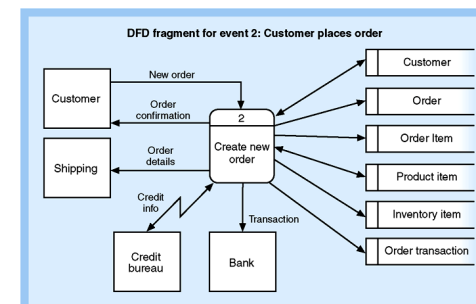
22

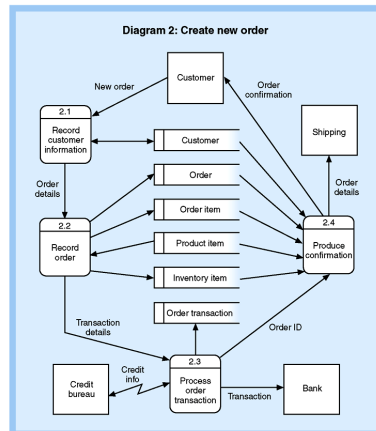## High-level Structure Chart for the Customer Order Program

MONASH University
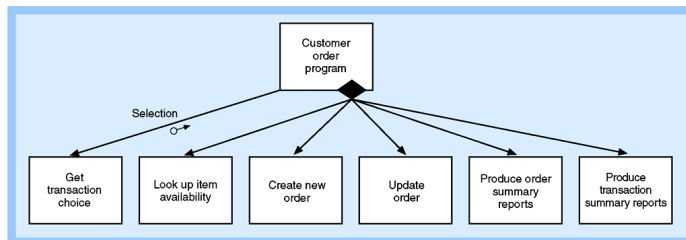
23

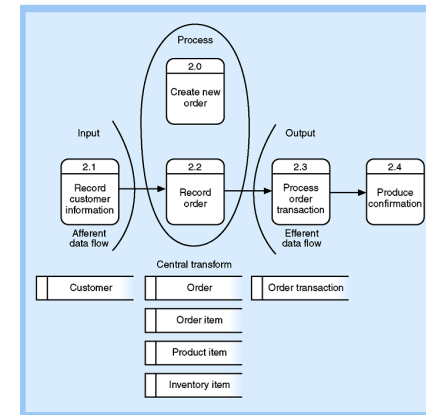## The *Create New Order* DFD Fragment

MONASH University

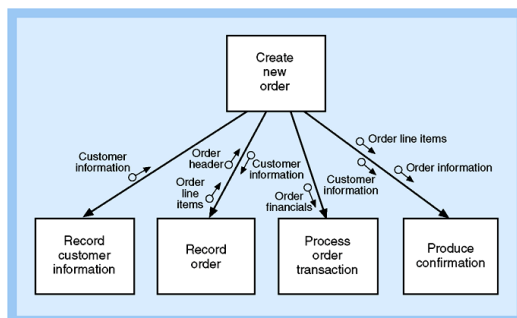
24

6

Exploded View of Create New Order DFD

Rearranged Create New Order DFD

First Draft of the Structure Chart

## Steps to Create a Structure Chart from a DFD Fragment (continued)

- Add other modules
  - Get input data via user-interface screens
  - Read from and write to data storage
  - Write output data or reports
- Add logic from structured English or decision tables
- Make final refinements to structure chart based on quality control concepts

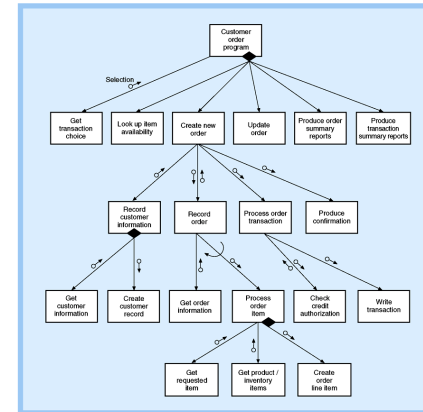## The Structure Chart for the *Create New Order* Program
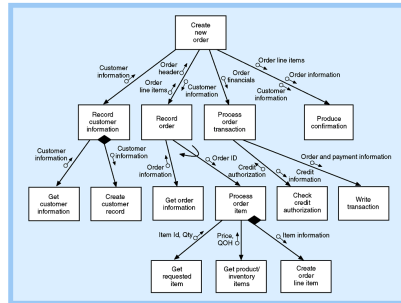
29

Combination of Structure Charts
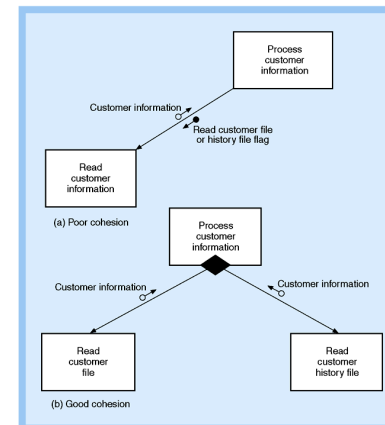

30

Evaluating the Quality of a Structure Chart

- Module coupling
  - Measure of how module is connected to other modules in program
  - Goal is to be loosely coupled
- Module cohesion
  - Measure of internal strength of module
  - Module performs one defined task
  - Goal is to be highly cohesive

31

Examples of Module Cohesion

32

8

Module Algorithm Design: Pseudocode

MONASH University

- Describes internal logic of software modules
- Variation of structured English that is closer to programming code
- Syntax should mirror development language
- Three types of control statements used in structured programming:
  - Sequence: sequence of executable statements
 - Decision: if-then-else logic
  - Iteration: do-until or do-while

33

Integrating Structured Application Design with Other Design Tasks

MONASH University

- Structure chart must be modified or enhanced to integrate design of user interface and database
  - Are additional modules needed?
  - Does pseudocode in modules need modification?
  - Are additional data couples needed to pass data?
- Structure charts and system flowcharts must correspond to planned network architecture
  - Required protocols, capacity, and security

34

Structure Chart Showing the Create New Order Program

MONASH University

35

MONASH University
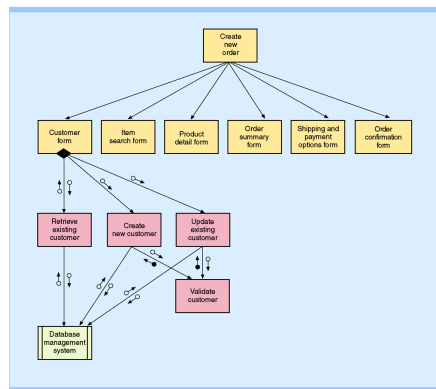
- Analysing Data with Excel

- See Module 11 Excel 2016
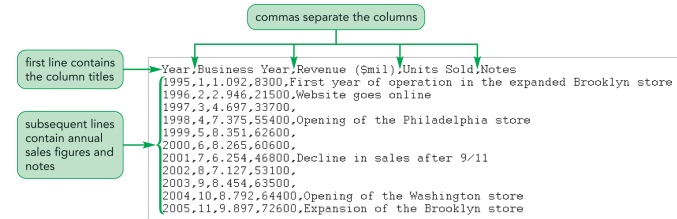
36

Writing a Data Query — MONASH University

- Text files
  - Simple, widely used format for storing raw, unformatted data (text and numbers)
  - Useful for sharing data across software programs and computer systems
- Understanding Text File Formats
  - Use a delimiter (space, comma, or tab) to separate columns of data
  - Use fixed-width text file to start each column at the same location

37

Writing a Data Query — MONASH University

commas separate the columns

first line contains the column titles

subsequent lines contain annual sales figures and notes

```
Year,Business Year,Revenue ($mil),Units Sold,Notes
1995,1,1.092,8300,First year of operation in the expanded Brooklyn store
1996,2,2.946,21500,Website goes online
1997,3,4.697,33700,
1998,4,7.375,55400,Opening of the Philadelphia store
1999,5,8.351,62600,
2000,6,8.265,60600,
2001,7,6.254,46800,Decline in sales after 9/11
2002,8,7.127,53100,
2003,9,8.454,63500,
2004,10,8.792,64400,Opening of the Washington store
2005,11,9.897,72600,Expansion of the Brooklyn store
```

38

Introducing Databases — MONASH University

- A database is a structured collection of data
- Databases are commonly used as the data sources for Excel workbooks
- A database is divided into separate tables
- Each table is arranged in columns and rows; also referred to as fields and records
  - A field stores information about a specific characteristic of a person, place, or thing
  - A record is a collection of fields
- Excel can retrieve data directly from most database programs

39

Introducing Databases — MONASH University

- Different tables are connected through database relationships; fields common to each table are used to match records in different table
- A one-to-one relationship is one in which one record in a table is matched to exactly one record from a second table
- A one-to-many relationship is one in which one record is matched to one or more records in a second table
- Relational databases are ones in which tables can be joined through the use of common fields

40

Introducing Databases

MONASH
University

each column
is a field

each row is
a record

StoreID	StoreAddress	StoreCity	StoreState	StoreZIP	StorePhone	StoreManager	StoreEst
Store01	4122 Coolidge Street	Brooklyn	NY	11232	(929) 555-0144	Valerie Wilburn	1995
Store02	45 Orange Avenue	Philadelphia	PA	19125	(267) 555-8901	Fred Mitchell	1998
Store03	744 James Lane	Washington	DC	20017	(202) 555-7814	Suzette Blake	2004
Store04	105A Bretain Drive	Chicago	IL	60622	(312) 555-0038	Nico DeJesus	2008
Store05	6133 Youngman Street	Indianapolis	IN	46202	(317) 555-9904	Renato Viera	2016

Record: 1 of 5   No Filter   Search

41

Exploring the Data Model

MONASH
University

- The data model is a database built into Excel that provides database tools
- Data model database contents are immediately available to PivotTables, PivotCharts, and other Excel features
- The data model is constructed from different tables related by common fields

42

Summary

- Program design with Structure Charts
- Analysing Data with Excel – Work through Module 11 on your own

- Note: You are expected to use Structure Charts for documenting your Assignment 2

- Next week – Introduction to Access 2016

MONASH
University