

BCI433 - IBM i Business Computing

**Week 1: Introduction to IBM i &
Writing CLLE Programs**

Agenda

- Welcome – Course introduction
- Course credits
- Intro to IBM i
- Definitions, Languages
- Write CLLE Programs
- Lab 1
- QuickCheck (Questions)

Lesson Objectives

The objectives of the lecture and lab 1:

- To familiarize you with basic IBM i (OS) operations in 5250 Emulator, i.e. green/white screen environment.
- Explain the definitions: objects, library lists, system values, and more
- Introduction to tradition IBM i development environment, and write/run your first CL program(s)
- Introduction to printing
- Install ACS and RDi on your PC

Welcome

- Welcome to BCI433
- Course introduction
 - On Blackboard
 - Course outline
 - Course website
 - IBM i 7.4 Documentations

IBM i

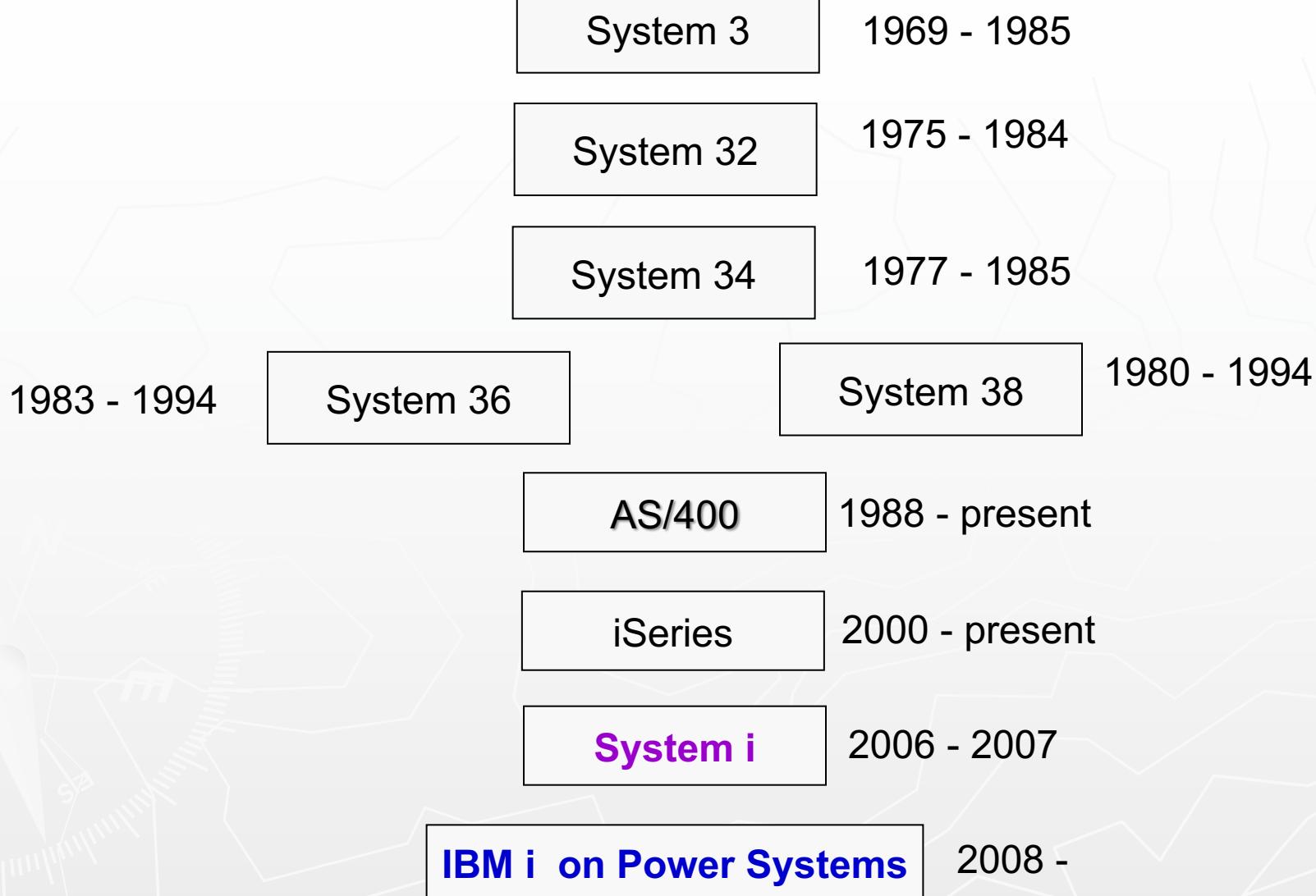
Developed by IBM to support medium to large scales business

i means iNTEGRATION!



A server designed for the on-demand challenges of Web and e-business, as well as core On-line Transaction Processing (OLTP) workloads, with support for multiple operating and application environments.

The History of IBM i



A Naming Nightmare

- The operating systems for IBM's midrange server line:
 - IBM i – runs on Power Systems or System i.
 - i5/OS (OS/400 V5R3)– runs on iServers (i5, i Series).
 - OS/400 – runs on AS/400.
 - CPF – runs on System 38.
 -
- IBM i, i5/OS and OS/400 are basically the same.

Power Systems

- In April 2008, IBM officially merged **System i** and **System p** under the same name – **Power Systems** with identical hardware and a choice of operating systems.
 - Being as System i, Power Systems run **IBM i**.
 - Being as System p, Power Systems run AIX or Linux.
- BCI433 covers mainly the System i part of IBM Power Systems, so we mix the usage of IBM i and Power Systems.

IBM i – A Business System

➤ Used by Banks, Retailers, Insurance companies..., e.g.

- McDonald's Canada
- Canadian Tire
- Kraft Foods
- Loblaws
- Holt Renfrew
- Coca Cola
- ToyRUs
- Casinos
- Lowes
- Staples
- Winners (TJX)
- Bell
- Electra/Hydro Utilities
- Bombardier
- City Government...
- more...

IBM i – Features

- RDBMS - DB2/400
- Menu Driven Interface
- Multiuser/Multitasking
- OS is Object Oriented
- Supports many Programming Languages
- PASE, QSHELL - complete layer ,which help to port anything from AIX to IBM i with minimal effort

Power System in Seneca

- Server name: ZEUS
- Installed: in January, 2009.
- Model: IBM Power 520 Express
- URL: zeus.senecac.on.ca

PowerVM Partitioning IBM AIX,
IBM i, and Linux on a Single
POWER6 System



Connectivity Tools

- Access Client Solutions ([ACS](#))
 - [5250 Emulator](#), Navigator for i
 - Production Environment
- Rational Developer for i ([RDi](#))
 - The Eclipse-based Rational IDE
- MochaSoft
 - 5250 Emulator
- Websphere Devbelopment Studio Client (WDSC)
(large install)

Let's sign on!

- Using IBM i ACS (5250 Emulator, Green/White Screen)
 - Find the Handouts file [ACS_Setup-2197.pdf](#), then follow the instruction to install ACS.
 - Your IBM i (Zeus) server userid/password can be found in the Grade Center on Blackboard



IBM 3486 Terminal with 5250 functionality, From Wikipedia

Commonly Used Function Keys

Function Key	Function
F1	Help
F3	Exit
F4	Prompt
F5	Refresh Screen
F9	Retrieve Previous Command
F12	Cancel
F23 (Shift+F11)	More options
F24 (Shift+F12)	More function keys

CL Commands

- Syntax: CL commands consist of a verb (an action), an object, and sometimes an adjective:
 - WRKACTJOB, DSPSPLF
- Some actions are:
 - WRK, DSP, PRT, DLT, CHG, RMV, ADD, EDT etc
 - WRK* will bring all commands using a wildcard “*”

CL commands (continued)

- Sign off: **SIGNOFF**
- Run a menu "Main": **GO MAIN**
- Send message "HELLO" to yourself, e.g.
DS433A35: **SNDMSG MSG(HELLO) TOUSR(DS433A35)**
- Display message: **DSPMSG**
- Run program "STRJOB": **CALL STRJOB**
- Run program "STRJOB" which is in library
QGPL: **CALL QGPL/STRJOB**

Objects

- Everything on the IBM i that has a name and takes up space in storage is an **object**, similarly like on Unix/Windows, everything is a file.
- On IBM i, objects have **types**. Common object types:
 - Library (*LIB)
 - Command (*CMD)
 - File (*FILE)
 - Program (*PGM)
 - Module (*MODULE)
 - Data area (*DTAARA)

Objects (continued)

- The object type determines what programs are allowed to act upon that object
- When you create an object, you will be its owner and you will have all the permission on that object.
- The **object types** used in Lab 1:
 - *USRPRF, *LIB, *CMD, *MSGQ, *OUTQ, *FILE and *PGM

Libraries

- **Library:** an object whose purpose is to 'store' and index other **objects**.
 - i.e. objects are 'stored' in libraries.
 - like a **directory** in Unix/Windows
- Object Type is *LIB
- QSYS is the only library that can contain other libraries
- Allowing access to an object additionally requires allowing access to the library that contains the object

About Your Student Library

- Your 'student library' is the library which has the same name as your **UserID** or profile.
- What Objects Do You Have (in your course library)?
 - An Output Queue which has the same name as your userid.
 - All objects which you create will be stored in your current library which is your student library by default.

Library List

- A library list is an ordered list of libraries with associated objects.
- Library List consists of 4 portions:
 - System portion. Always at the top and contains IBM system libraries like QSYS, QHLPSYS, QUSRSYS, etc. The system portion can hold up to 15 library names. Names start with "Q" or "#"
 - Product portion. Optional
 - Current library, which is your student library. Associated with a user profile and there can only be one in the list at a time.
 - User Portion of library list, which is QUSRLIBL and usually contains libraries with commonly shared user objects
- A system administrator decides what libraries are included
- Library Lists are built when you sign on and are deleted when you signoff

System Values

- Variables maintained by the operating system to set up IBM i.
- Examples:
 - **QSYSLIBL** – System libraries for library list
 - **QUSRLIBL** – User libraries form library list
 - **QDATE**
- The command to display a system value:
DSPSYSVAL QSYSLIBL

User Profiles

- Each **UserID** has a **User Profile** which describes the user and user's authorities
- User Profile Contains information such as
 - Current Library,
 - default output queue and message queue
 - password,
 - class of user

Work With Active Jobs

- A **job** is any and every piece of work on the IBM i.
 - Jobs run in subsystems rather than directly in IBM i
 - Two types:
 - ▶ interactive jobs – start when a user signs on
 - ▶ batch jobs – background, e.g. compile a program
- Command **WRKACTJOB** shows you:
 - all the jobs that are currently running in IBM i
 - jobs' status.
 - the option to end jobs

Output Queues & Spooled files

- A **queue** is a line-up! A place (object) where things wait.
- Examples of IBM i queues:
 - **job queues**: where batch jobs wait
 - **message queues**: where messages wait
 - **output queues**: where spooled files wait to print
- **Spooled files**: Formatted output (in output queue) ready for printing

Source Physical Files

- Source physical file – object that stores program source code
- Object type: *FILE
- A source file may have many members.
- Each member of a source physical file is a separate program (code).
- When a member, e.g. a CLLE code, is successfully compiled a new program object (*PGM) is placed in a library (usually your current library)

PDM

- Program Development Manager (used in green/white screen)
- The tool that gives easy access to a programmer's stuff (libraries, files, and members)
 - STRPDM
 - WRKOBJPDM DS433A35
 - WRKMBRPDM QCLLESRC
- "WRK", i.e. work
 - What is the Unix/Linux command or English word that is equivalent to "WRK" in IBM i?

Creating CL Program

- About CL
 - What does CL stand for?
 - CL program type: CLLE
 - CL Code is between : PGM and ENDPGM
- To list all members in source physical file QCLLESRC:
 - [WRKMBRPDM QCLLESRC](#)
- To work with compiled CL program - *PGM object:
 - [WRKOBJPDM DS433E35](#)
- To see your compiler listing
 - [WRKSPLF](#), or
 - Use Printer Output from ACS – in pdf

LPEX / SEU Editor Commands

Command (on line #)	Operation
I	Insert a line
D	Delete a line
C	Copy a line
M	Move a line
A	Paste After
B	Paste Before
CC, CC	Copy multiple lines
DD, DD	Delete multiple lines
MM, MM	Move multiple lines

Lab 1 demo

Homework

- Review lecture notes.
- Setup ACS on your PC
- Complete Lab 1
- Install RDi 9.6 (for Lab 2 in next week)



The End