Systems Analysis and Design

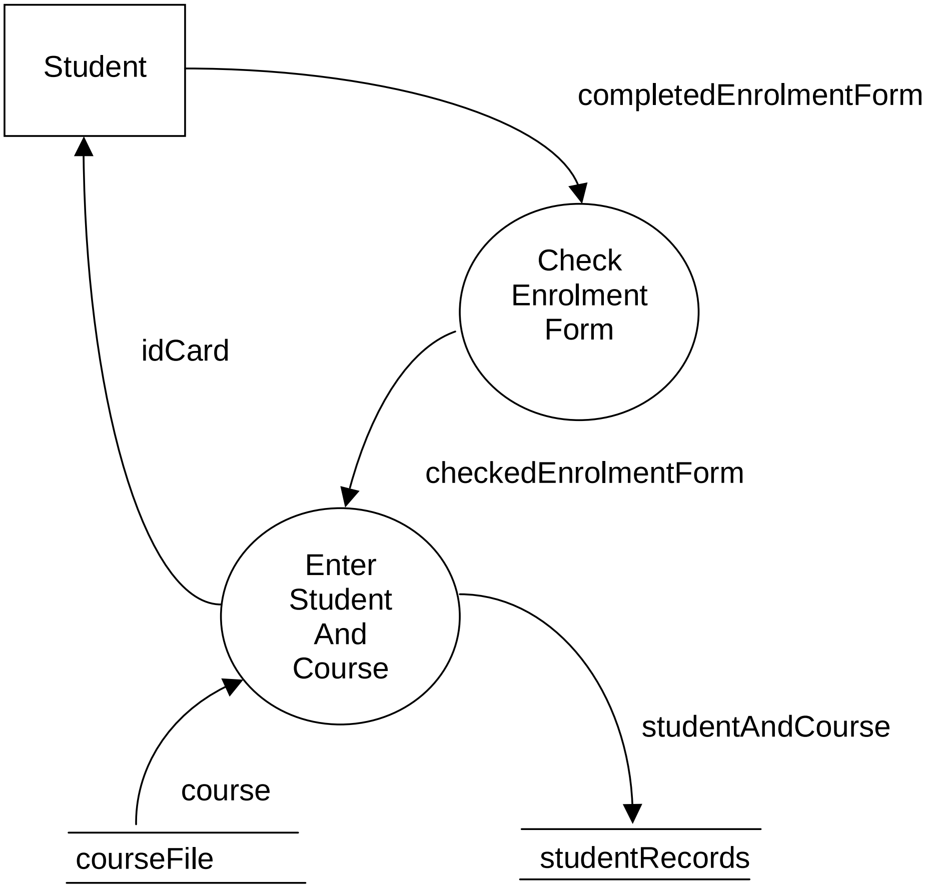
7 Data Dictionaries

Terry Marris December 2007

A data dictionary describes the data items found in data flow diagrams and entity relationship diagrams. It provides a starting point for developing screen layouts, printed reports and programming logic. It provides an opportunity to detect design errors. It is maintained and expanded throughout the entire systems analysis and design process. We look at what a data dictionary is and how to construct one.

7.1 Data Flow Diagram

Data dictionaries are derived from data flow diagrams.



A student completes an enrolment form, has it checked and signed by a lecturer, and hands it to the enrolment clerk for processing. The enrolment clerk checks the course file for the requested course and issues a Student Id Card. The student and course details get placed on the student records file.

7.2 Data Dictionary

Data Dictionary

completedEnrolmentForm = studentId
+ studentName
+ studentDateOfBirth
+ courseCode
+ courseEndDate

studentId = integer(6)
The first two digits represent the year of enrolment e.g. 07 for 2007. The remaining four digits uniquely identify the student

studentName = studentFirstName
+ studentSurname

studentFirstName = char(15)
up to 15 characters are allowed

surname = char(15)
or family name. Up to 15 characters are allowed

studentDateOfBirth = short Date format e.g. 16May1989
year must be before 1989 i.e. they must be at least 16 years old on the 1st September when they enrol

courseCode = char(5)
up to five characters e.g. A0371 = Access to Computing course
Every courseCode has associated with it a courseTitle

courseEndDate = short Date format e.g. 30Jun2008
the date the person ceases to be a student

checkedEnrolmentForm = enrolmentForm
+ lecturerName
+ dateLecturerSigned

lecturerName = initials
+ surname

initials = char(3)
up to three characters allowed

CourseFile = { courseRecord }
CourseFile contains zero, one or many courseRecords

courseRecord = courseTitle
+ courseCode

courseTitle = char(25)
up to 25 characters e.g. BTEC Nat Dip Comp Studies

And so on.

You use

=	to mean *consists of*
+	to mean *and*
{ }	to mean a *repeated item*
[ ]	for an *either/or* situation e.g. gender = [male, female]
( )	for an *optional* item

You create a data dictionary top down, starting with the most general, working towards the fundamental data types:

- character - letters of the alphabet, digits, punctuation symbols
- integer - whole numbers such as 1, 4, 7, 11
- float - numbers with a decimal point such as 3.1416, 7.89
- Boolean - true or false.

You include comments relating to:

- synonyms - other names for the same object
- data type
- length e.g. 15 characters allowed for name
- validation rules e.g. dateOfBirth must be before 1 September 1998.

You add corrections and more detail and as you progress through the analysis and design. You order the entries, for example, alphabetically.

As you can see, data dictionaries end up being rather large. You use a word processor, or a spreadsheet, or specialised software to maintain them. All the time you are looking to remove all possible errors.

Bibliography

PRESSMAN R.S. *Software engineering - A Practitioners Approach* 3rd Ed McGraw-Hill 1992
KENDALL & KENDALL *http://www.mgt.ncu.edu.tw/~ylchen/sasd-slide/chap10.ppt* accessed 18 Dec 2007
WEISERT C *www.idenews.com/life-cycle/datadefn.html* accessed 18 Dec 2007