

Data Flow

Monday, 5 August 2019 10:39 AM

Intro

The syllabus requires us to consider process-modelling tools
- one such tool is "DATA FLOW DIAGRAMS" or **DFD**'s for short.

This Onenote resource is inspired and enriched by resources provided by Ben Harvey, with thanks.

DFDs DO

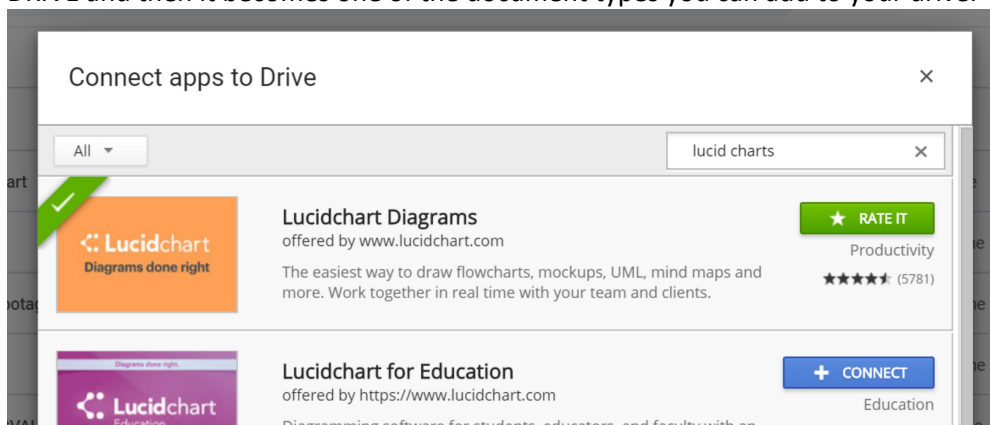
- explore interactions between user and system
- track data through processes
- map interactions with data stores
- have DEFINITE rules for construction
- have a couple of different display standards

DFDs NO NOT

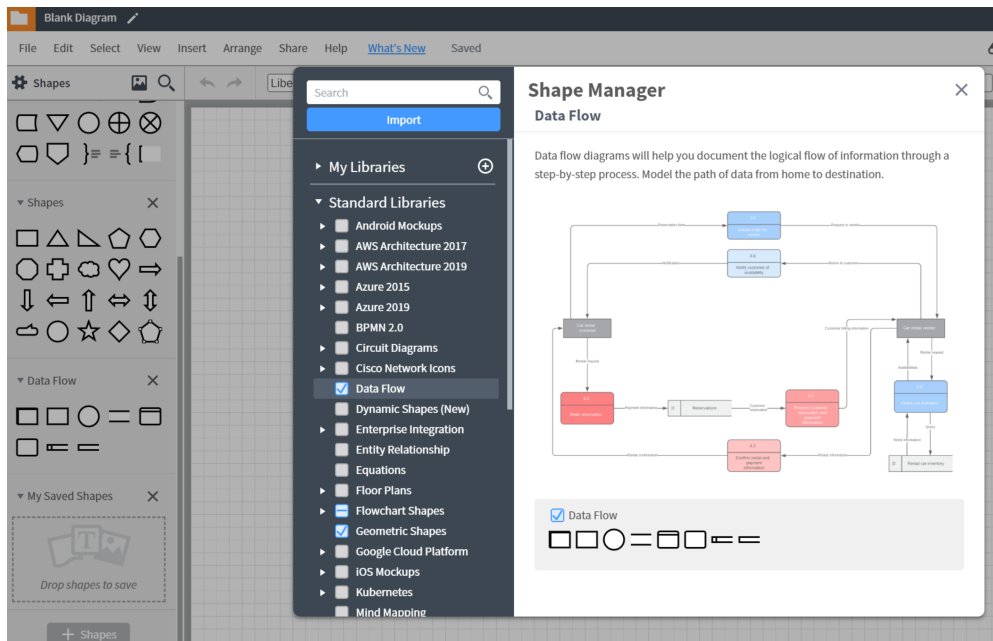
- express HOW a process works, merely what goes in and what comes out
- directly map to code, database design or system architecture

Tools

I find LUCIDCHARTS has a good range of DFD symbols and connectors. You can add it into your GOOGLE DRIVE and then it becomes one of the document types you can add to your drive.



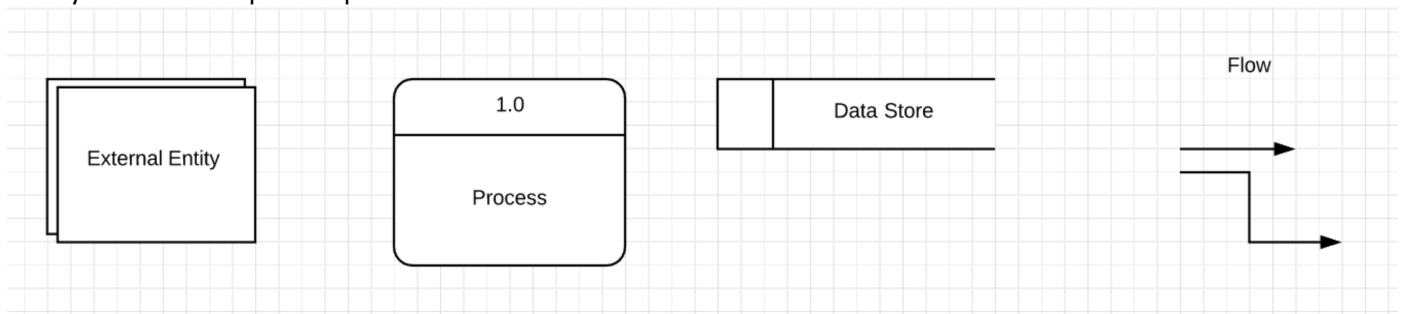
You need to ADD the Dataflow SHAPES to your sidebar



other drawing packages I have played with (including draw.io, google drawings, balsamiq, gliffy and many more) have a frustratingly limited set of symbols that seem all to exclude the core DFD symbols, Lucidcharts has the right symbols out of the box, and deals natively with your google drive. Win-Win.

Components of a DFD

The symbols are simple and powerful



There are variants, but the ones the syllabus uses are these so we will too.

ENTITIES - these are *sources* or *destinations* of a data flow that is outside the described processes

PROCESSES - these *transform* incoming data into the outgoing data flow

STORES - repositories of data, come in 3 common flavours: "**D**" indicates a permanent computer file, like a database; "**M**" indicates a manual file; "**T**" represents a transient store that is deleted after processing

FLOW - these are connectors, track the flow of data between the components

Drawing a DFD

We (generally) draw a DFD in a number of steps - we use a TOP DOWN approach - this means we start with a GENERAL overview, then subsequent diagrams further break down the general.

Top level is often called "context level" and is our first port of call...