Enterprise Resource Planning

TD 1: Towards integrated software

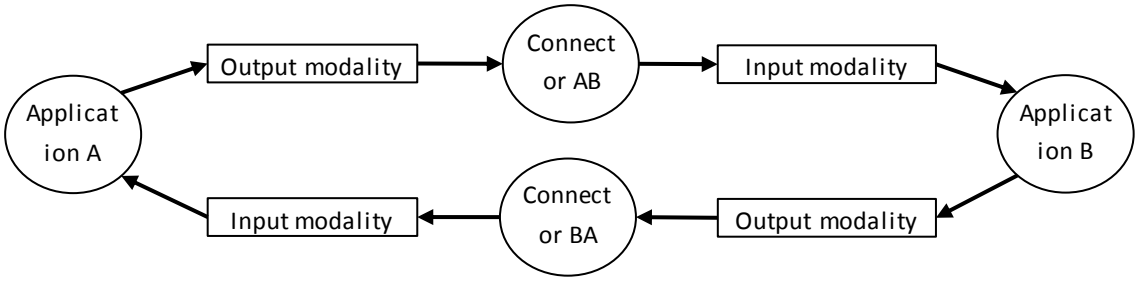
The management of the back-office of an enterprise needs different software applications. Some of these applications are often merged in bigger, and bigger, applications. In this work, we try to understand some technical reasons that lead to go for more integrated software.

Exercise 1: The activities of the enterprises

Recall some classical activities inside the companies. For each activity, give the name of a software solution which exists to manage it.

Exercise 2: How to connect two applications software

The aim is to allow the exchange of data, in both directions, between two enterprise applications, which were made to work together. To do so, some extra programs are written to translate data (stored in files, databases, flows…) in order to make possible these exchange of data between enterprise applications. These programs are called *connectors*. These small programs are often some BATCH, POWERSHELL, SHELL, PHP, Python… scripts, which are first going to read/extract/receive data (according to the modality used) from the first software, then analyze, treat, and translate it, and then read/load/send (according the modality used) the result of the conversion to the second software.



The main input and output modalities are the following:

1. File resource: Read / Write the file (csv, xml, …)
2. Database resource: Extract / Load with the database
3. Flow resource: Send / Receive a message via a webservice (XML-RPC, …)

In an information system, the triggering of a connector can be made:

- Once and manually by the user
- Once and automatically by a program (which, by example, is waiting for an event, stored procedure (*trigger*), …)
- Periodically and automatically, every X seconds, each hour, each night… by the Operating System (with the CRON system under Unix/Linux system, or Planification Tool under Windows) or by internal programming of the connector (infinite loop).

Question 1: By considering only the three modalities presented above, draw schematically the 9 possible kinds of AB connectors between the two applications. Precise each time, by a verb, what work is done by the connector on the intermediate resource among the connector and the application.

Question 2: Considering only the three modalities presented above, how many configurations are possible to connect, in both directions, the applications A and B.

Exercise 3: Connecting several applications in point-to-point mode

We are going to study the number of connectors necessary to make several applications exchanging data together. For that, let's recall some graph theory to model the problem:

- The software infrastructure is represented by an oriented graph, where the vertices are applications software, and the edges are the connectors (allowing to transmit data in only one way).

We consider the graph of software infrastructures as a complete graph.

Note: a complete graph has all its vertices linked to all others.

Question 1: Draw the complete graphs modeling the cases of 2, 3 and 4 applications software.

Question 2: How many edges are present in these graphs?

Question 3: Draw again the case of 4 applications, and add a fifth application software. How many new edges must be added to keep the graph complete?

Question 4: Deduct (from the previous question), the rule of the number of new connectors needed to create when adding a new application software (and which will have to be installed and maintained)?

Question 5: For an information system with N software application, how many connectors must be created, installed and maintained to make all these applications communicate with each others?

Exercice 4: Integrated approach

In an integrated approach (EAI: Enterprise Application Integration), an extra application software play to role of a hub. Then, every software communicate with others only by sending data through the hub. Thus, each application has to be connected to the hub. We then call this architecture to be star-shaped.

Question 1: Draw the graphs modeling the cases of 2, 3 and 4 applications software.

Question 2: How many edges are present in these graphs?

Question 3: Draw again the case of 4 applications, and add a fifth application software. How many new edges must be added to keep the graph complete?

Question 4: Deduct (from the previous question), the rule of the number of new connectors needed to create when adding a new application software (and which will have to be installed and maintained)?

Question 5: For an information system with N software application, how many connectors must be created, installed and maintained to make all these applications communicate with each others?

Question 6: An integrated infrastructure is a complex project and need a big investment at the beginning. Do you think this kind of architecture can be deployed in small information systems with only few applications software? Do you think that ERP software is rather easier or harder to install, compared to several application connected through EAI?