



Learning Objectives

At the end of this module, the student should be able to:

1. Define System Integration.
2. Distinguish the different types of System Integration methods.
3. Enumerate the advantages and disadvantages of System Integration.

SYSTEM INTEGRATION

System integration is the process of linking together different systems (components) to functionally cooperate as a whole. System integration can be mono-directional (one-way) or bi-directional (both ways). In short, data and functionality should flow seamlessly between integrated systems, communicating optimally with each other, and thus reducing complexity.

System Integration is the process of integrating all the physical and virtual components of an organization's system. The physical components consist of the various machine systems, computer hardware, inventory, etc. The virtual components consist of data stored in databases, software and applications. The process of integrating all these components, so that act like a single system, is the main focus of system integration.

System Integration makes it possible for new technologies such as cloud, big-data, and IoT to communicate quickly with each other and current IT systems, so organizations can retrieve value from investments rapidly. With a good integration platform, the time span of even the most complex integration project can now be reduced from several years to a few months.

System Integration Challenges

There are a number of issues that make System Integration challenging for all organizations. According to certain studies, up to 70% of all integration projects fail in some aspect. This is probably more common for complex projects, but sometimes even relatively simple System Integration projects can run into trouble. Most of the failures are not due to the chosen integration technology itself or technical difficulties with the systems in the scope, but due to project and change management issues.

Typical reasons for System Integration project failure may include:

- **Constant Changes of the Integration Landscape**

The longer the project takes, the more significant this issue becomes. To manage this risk, time is of the essence, keeping the integration projects short improves the success rate of the project. Further, an agile working methodology that can cater to changing requirements along the way and also after the project is essential for the success of the System Integration.

- **Lack Of Skilled Resources**

System Integration requires expertise that is not easy to come by. Having excellent integration technology is not enough if the required expertise is not there. Most companies struggle to find and retain employees with the required skills set for System Integration. The best way to tackle this issue



is to use an external third-party provider that can bring the needed integration expertise into the table as required, in addition to providing the integration technology.

- **Lack of Accountability**

When integrating many different sub-systems, the accountability for the success of the integration becomes blurred very easily. It may have multiple stakeholders (e.g., vendors, system owners, etc.) in the equation, none of whom is responsible for the entire System Integration. They only care and handle their side of integration at best, but they will not venture outside their own territory.

But integration always has more than one party. So, when something goes wrong, the situation turns very easily to finger pointing and blaming the other parties, instead of someone “owning” the integration. If a single party handles the System Integration project, that party is also (often contractually) responsible for the success of such a System Integration project, and there is no ambiguity over accountability.

Advantages of System Integration

1. Expanded profitability

It supports the organization’s efficiency massively. Working with various systems and projects takes up a significant measure of time, compelling the representatives to zero in on tedious manual activities instead of on projects within reach.

2. Better administration and examination

A high number of projects make it intense to dissect just how an organization works. Therefore, running various investigations for each program isn’t simply tedious, it’s regularly pointless.

3. Lower cost

The expense of keeping up various systems can be fairly high. Each program requires certain support, which may be done on various occasions. Planning support and overhauls can be tedious and exorbitant. Hours spent on all-out help may surpass the hours the whole system is working.

4. Improved consumer loyalty

System Integration permits accomplishing better consumer loyalty by diminishing the time spent on giving items and administrations. Whereas, working with different systems and projects can take up a huge chunk of time.

Disadvantages of System Integration

Even though System Integration is by all accounts an astounding answer for some issues identified with utilizing various software, it has a few disadvantages.

1. Security problems

Having a few projects is superior to one incorporated system. Any hack or misrepresentation could gain admittance to the entirety of the information instead of one piece of it. System Integration makes data more defenceless than it used to because of the channels through which information streams starting with one program then onto the next.



2. Complex updating

IT group is probably going to make some extreme memories. Integration doesn't transform a few projects into one, it simply integrates them. Projects actually should be overhauled independently. The interaction is unpredictable and tedious.

3. Significant expense

Paying for each program independently is needed. Once in a while, costs are high to the point that an entrepreneur may begin contemplating custom program advancement instead of System Integration.

System Integrator's Role

A Systems Integrator ("SI") is an individual or a business/company that specializes in implementing, planning, coordinating, scheduling, testing, improving and sometimes maintaining IT systems. The role of the systems integrator is narrowed down to enabling the data integrations between the different existing systems of the end customer defined in the project scope. This may mean anything starting from simple internal point-to-point connections to very complex many-to-many integrations both internally and with third parties.

Good examples of SIs are, e.g., Deloitte, IBM, Accenture, TCS, etc. They deliver large IT projects (e.g. ERP projects) trying to manage such projects and the numerous suppliers involved.

The systems integrators role is normally to design, implement and test the integration solution, but the role of the systems integrator may also include continuous management of the solutions as well as contacting third parties to enable connections with them. Most importantly, however, the systems integrator brings to the table integration expertise that the customer is otherwise lacking internally (or has a shortage of available internal resources at hand).

System Integrator's Responsibilities

It entails enabling the data integrations between the various existing systems of the final customer as defined by the project scope. This can include anything ranging from simple point-to-point connections to very complex many-to-many connections both internally and with third parties.

The task of System Integration at this point usually begins with a meeting with the client aimed at assessing the client's business needs and defining the technical requirements for a system or collection of systems that meet those needs. Only then can the systems integrator supply the most appropriate modules and configuration to ensure smooth System Integration.

The role of the system integrator usually covers all aspects of design, implementation and testing of the solution offered by the company. In some cases, the systems integrator may also be involved with maintenance in terms of on-going management of the solutions as well as contacting third parties to enable connections with them. Most importantly, however, the systems integrator brings with him the integration expertise that the customer is otherwise lacking internally.



Types of System Integration

Given the zone of utilization and sort of utilization, integration administrations can be separated into these different classifications:

- **Enterprise Application Integration (EAI)**

Enterprise Application Integration is a help-based integration. It's an interaction that speaks with various administrations, assembles information and afterwards continues with additional means dependent on wanted activity or a work process. The cycle can be set off with uncovered help.

- **Data Integration (DI)**

At the point when you need to merge your administrations into one main issue of access, you need information integration. For the most part, there are loads of various information sources (or data sets) and you need them in one spot of access, all merged.

- **Electronic Document Integration/Interchange (EDI)**

EDI (Electronic Document Integration/Interchange) is a central business to business situated interaction. Its capacities are for the paperless trade of reports and electronic regulations.

- **Legacy System Integration**

Legacy system integration allows organization to connect legacy systems with other, newer systems.

- **Third Party System Integration**

Building custom software is not feasible in many cases. Instead, third-party software products may be used, which may need to be integrated with other systems and applications already in use in the organization.

- **Business-to-Business Integration**

This type of integration allows businesses, for example across a supply chain, to easily exchange data and knowledge, reducing redundancy.

- **Big Data Integration**

Essential in any big-data project is the integration of data origination from disparate business systems, various sources, and different formats. It allows all data to be combined in a single view to be used for reporting and analysis.

- **Cloud Services Integration**

Connecting various applications, systems, and databases to be integrated in cloud services, enabling access by multiple devices over a network or the internet.

Common Data Format Integration

Integrating different IT systems to each other usually require that the data coming out from one system needs to be transformed to a different data format used by the receiving system. As with the Star Integration, if each transformation needs to be done system-by-system basis, the number of data transformations increases significantly and becomes a high maintenance task.



To overcome this problem, common data format approach, allows each system to do only one data conversion from its native format to the common (and vice versa). This way the number of required data transformations is just as high as the number of the sub-system.

Sample: Integration of Tools Supporting Everyday Work

Many tools are used at work (e.g., Google Docs, emails), the integration of which can bring many benefits. It helps save time and eliminates repetitive tasks that often make you tired. Fortunately, there are tools on the market that allow for this type of integration, and one of them is Zapier.

Zapier: A Short Definition

Zapier is a handy tool. It allows you to automate work by linking your apps so that they share data. You can use it for Slack, Dropbox, Asana, Gmail, and more. While the application data is seemingly unrelated, Zapier helps create chains of connections, making the company's day-to-day tasks much easier.

Zapier at Studio Software

One example is linking Gmail to CRM by using Zapier, which has automated the process of adding a contact to CRM. All necessary data from potential customers are now automatically saved in CRM, so you don't have to do it manually. Zapier won't solve every integrational problem you struggle with, so sometimes it may be better to implement the integration using the systems' APIs.

System Integration Process

Successful integration needs good preparation. Even relatively simple projects can sometimes become difficult and complicated. These activities require expertise, and we always create a detailed plan with a step-by-step checklist to make it successful.:

1. Analyzing the API documentation of both products. Checking the documentation allows us to see how the API data works, what protocols it uses, and what possibilities it offers (e.g., if it will notify itself when it performs an action or repeats the action in case of an error).
2. Checking test environments (sandbox) and obtaining separate access data for these environments. Each website should provide a so-called sandbox for testing—a place where we can easily test the entire system operation (e.g., make a test payment, or connect a test payment card).
3. Creating separate configuration files in your application for specific environments (e.g., dev, stage, or prod). The local or development environment should use sandbox access data (URL, login, and password), and the configuration file in the production environment should contain data production.
4. Large-scale testing. Before the system goes live, we always carry out complex testing to eliminate potential errors. Making and following a checklist is very important because it is easy to forget to check various details. Testing different scenarios is a great way to check every possible situation and improve the quality of work.



References:

<https://www.youredi.com/>

<https://dynamics.folio3.com/>

<https://studiosoftware.com/>