



# THEORY OF INTEGRATION

---

## THE INDIRECT APPROACH

---

### ABSTRACT

---

In this paper, I will discuss something that I call *the indirect approach*. The concept of an indirect approach applies in the practice of translation and integration. We will also see the indirect approach apply to architecture that we discuss in other papers. The particulars of how the concept of the indirect approach is applied are different in integration, translation and architecture, but the concept is fundamentally the same.

# THEORY OF INTEGRATION

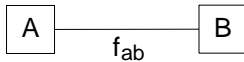
## THE INDIRECT APPROACH

---

### DIRECT VERSUS INDIRECT

---

Suppose there are two things that we wish to integrate. We will draw them as boxes and call them A and B as shown. You can think of the boxes as applications or documents. You can think of them as physical things like engines if you like. It really doesn't matter.

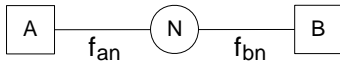


*Figure 1 Two systems directly integrated*

We will call these boxes *representations* or *systems*. Integration is the process of translating a representation from one form to another. The most direct means of integrating two representations is to create a single function such that

$$A = f_{ab}(B) \quad B = f_{ab}(A)$$

When we deal with integration, we always begin with two systems or representations, so the direct method seems most logical.



*Figure 2 Two systems indirectly integrated*

The alternative is an indirect means of normalization in which we create two functions such that

$$N = f_{an}(A) \quad A = f_{an}(N) \quad N = f_{bn}(B) \quad B = f_{bn}(N)$$

To integrate, we must apply two equations

$$A = f_{an}(f_{bn}(B)) \quad B = f_{bn}(f_{an}(A))$$

This is not so abstract as it may first appear. We think of the boxes as applications and the functions as application programming interfaces (APIs). You could, however, think of the boxes as documents and the functions as translators. Suppose for instance that A is Armenian and B is Bantu. Function  $f_{ab}$  represents an individual who speaks both Armenian and Bantu and can translate both ways. Our first inclination may be to find such a person, but our chances of finding such a person are remote. Now suppose that N is English. Function  $f_{an}$  represents someone who speaks Armenian and English. Function  $f_{bn}$  is a person who speaks Bantu and English. With two translators our integration can be achieved.

There is a distinct disadvantage to the indirect approach when dealing with just two systems. The direct approach requires only one function while the indirect approach requires two. If we add a

third system C, we see no advantage to either the direct or indirect approach. They both require three functions.

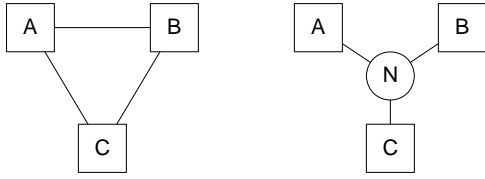


Figure 3 Three systems integrated directly (left) and indirectly (right)

With the direct approach we employ the functions:

$$A = f_{ab}(B) \quad B = f_{bc}(C) \quad C = f_{ac}(A)$$

While the indirect approach employs the functions

$$A = f_{an}(N) \quad B = f_{bn}(N) \quad C = f_{cn}(N)$$

It is only when we introduce the fourth element that the advantages of the indirect approach over the direct approach begin to appear.

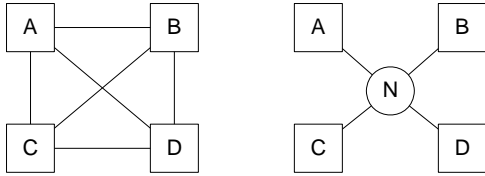


Figure 4 Four systems integrated directly (left) and indirectly (right)

In order to implement the direct approach, we need six equations:

$$f_{ab} \quad f_{ac} \quad f_{ad} \quad f_{bc} \quad f_{bd} \quad f_{cd}$$

Whereas, with an indirect approach, we only need four equations

$$f_{an} \quad f_{bn} \quad f_{cn} \quad f_{dn}$$

The complexity of the direct approach increases exponentially, whereas the complexity of the indirect approach increases linearly. For every element  $i$  added to a system using direct integration,  $i-1$  functions must be added. For example, when the tenth element is added to the system, nine new functions are needed. When the 100<sup>th</sup> element is added, 99 new functions are needed. Within a system using an indirect approach to integration, however, only one new function is needed for new element added.

Table 1 shows a comparison of the complexity between using a direct approach and an indirect approach of integration. For just two elements, the indirect approach is twice as complex as the direct approach. With three elements, they are equally complex. In a system of five elements, a direct approach is twice as complex as an indirect approach. In a system of 101 elements, a direct approach is 50 times more complex than an indirect approach.

| elements | direct          | indirect | ratio       |
|----------|-----------------|----------|-------------|
| 2        | 1               | 2        | 1:2         |
| 3        | 3               | 3        | 1:1         |
| 4        | 6               | 4        | 3:2         |
| 5        | 10              | 5        | 2:1         |
| 6        | 15              | 6        | 5:2         |
| 7        | 21              | 7        | 3:1         |
| 8        | 28              | 8        | 7:2         |
| 9        | 36              | 9        | 4:1         |
| 10       | 45              | 10       | 9:2         |
| 11       | 55              | 11       | 5:1         |
| 100      | 4950            | 100      | 99:2        |
| 101      | 5050            | 101      | 50:1        |
| x        | $(x^2 - x) / 2$ | x        | $(x-1) / 2$ |

Table 1 Comparison of complexity of direct and indirect approach

---

### INTEGRATION OF THE UNANTICIPATED

---

The most important concept here is not the math, but rather that in a direct approach to integration, every instance must be anticipated. For instance, the system architect must anticipate element G, that is, the seventh element, in order to create the functions

$$f_{ag} \quad f_{bg} \quad f_{cg} \quad f_{dg} \quad f_{eg} \quad f_{fg}$$

In the direct approach, the integrator must know both system A and system G. In an indirect approach, the architect is required to make no such anticipation. The function  $f_{an}$  does not have to anticipate the function  $f_{gn}$ . The new function may be applied without any change to the old one.

$$A = f_{an}(N) \quad N = f_{an}(A)$$

$$G = f_{ag}(N) \quad N = f_{ag}(G)$$

$$G = f_{gn}(f_{an}(A))$$

$$A = f_{an}(f_{gn}(G))$$

Thus, an indirect approach allows the integration of the unanticipated.

---

### DISRUPTION OF CHANGE

---

The impact of small changes to a system using direct integration becomes catastrophic in their consequences. Suppose, for instance, that we have a system of 10 elements. There are 45 interconnections or functions to be maintained. Each element within the system employs 9 connections or functions. Now suppose that one element changes. The system requires the updating of 9 functions in response to the change in one element. Let us further suppose that only one element can change within a given period T and that we have the resources to make 9 function changes within period T. Under these assumptions, we have the ability to maintain the system indefinitely.

Let us now compare the disruptions change to a direct system with that of an indirect system. Suppose that disruption increases linearly at 10 percent, that is, that within any given system 10 percent of the elements change within a given period of time T. With a directly integrated system, we have seen that 9 connections must change. Suppose, as we did before, that we have resources enough to make 9 connection updates with period T. With those resources, we can maintain a system of 10 elements directly connected. In an indirectly integrated system, each element has only one connection. We can maintain a system of 90 elements.

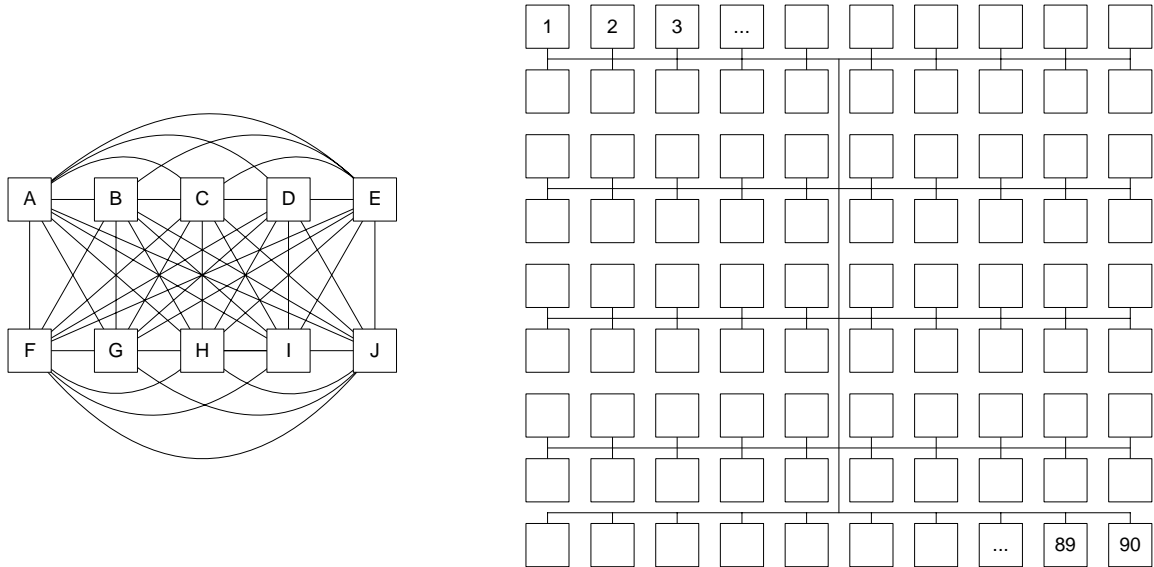


Figure 5 The resources needed to maintain these two systems are equivalent

Now lets disrupt our world. Suppose that we add a new element to the system without adding to our resources for maintaining the system. (Not that this could ever happen within an IT department!) Now we have 11 elements with 55 connections, or 10 connections connected to each element. Now every time an element changes we must update 10 connections, rather than 9. However, because we have not increased our resources, we can only change 9 connections within a given period T. At the end of every period there is another connection that we have not been able to update. Within five periods, there are five connections out of date or incorrect. In other words, 10 percent of the connections within system are outdated or dysfunctional.

---

### GROWING THE SYSTEM

---

Growth of a system is often viewed as a form of disruption. IT departments are consistently faced with growth of their network systems and network applications. Let us compare our ability to grow a directly integrated system and an indirectly integrated system.

As stated earlier, with nine resources we can maintain 10 elements in a directly integrated system and 90 elements in an indirectly integrated system. One resource is able to update one connection in a given period. Suppose that we are able to grow our staff by one resource per period. Let us continue to assume that 10 percent of the elements change every period. In an indirect approach, each element has only one connection, and since one in 10 elements change in a given period, each resource is able to support 10 elements. We should be able to grow an indirect system by 10 elements each period indefinitely. That is, the system supports linear growth.

|                   |    |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
|-------------------|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| period            | 0  | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | ... | 20  | 21  | ... | 27  |
| staff resources   | 9  | 10  | 11  | 12  | 13  | 14  | 15  | 16  | 17  | 18  | ... | 29  | 30  | ... | 36  |
| direct elements   | 10 | 11  | 12  | 13  | 14  | 15  | 15  | 15  | 15  | 15  | ... | 15  | 16  | ... | 20  |
| indirect elements | 90 | 100 | 110 | 120 | 130 | 140 | 150 | 160 | 170 | 190 | ... | 290 | 300 | ... | 360 |

*Table 2 Ability to support growth in direct and indirect systems*

Growing a directly integrated system is much more problematic. For elements 11 to 15 we are fine, but once we reach the 16<sup>th</sup> element in the system we have a problem. Remember that 10 percent of the elements change each period. With 16 elements, more than one element is changing each period. To avoid having to deal with probabilities, lets jump ahead to a system of 20 elements. With 20 elements, 2 elements are changing each period and each element has 19 connections. That means we need a staff of 36 resources to support the system. Note that it takes more resources to support the system than there are elements in the system.

The math for calculating the number of resources required is straightforward.

$p$  : percentage of resources changing in a given period

$x$  : number of elements in system

$f$  : number of connections in system

$$f_i = x$$

$$f_d = (x^2 - x)/2$$

$R$  : number of connections changing per period

$$R_i = px$$

$$R_d = px(x - 1)$$

For given resources  $R$ , we can support  $R/p$  elements in an indirectly integrated system. Since  $p$  is a percentage and thus less than or equal to one, the number of elements will be more than the number of resources. A directly integrated system requires  $R(R/p - 1)$  resources to support the same number of elements. For instance, if 20 percent of the elements change in a given a period, then a staff of 20 resources can support 100 elements in an indirectly integrated system.

$$x = R/p = 20 / 0.2 = 100$$

In a directly integrated system, that many elements under the same conditions would require nearly 50,000 resources.

$$R = p(x^2 - x) = (0.2) ( 100^2 - 100) = 49,900$$

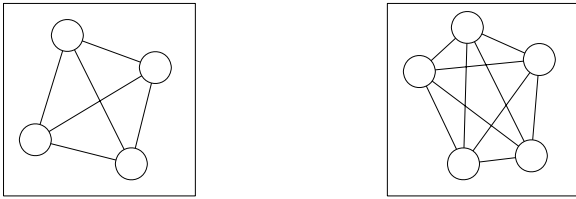
In other words, the directly integrated system is not sustainable in any practical sense.

---

## EMPLOYING THE CONCEPTS

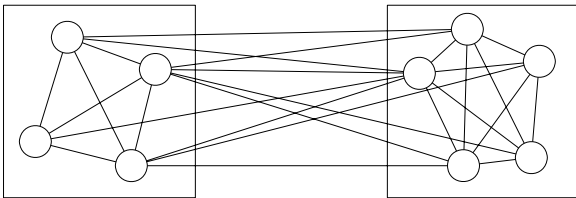
---

When integrating systems, applications or even organizations, we are never dealing with a single amorphous block. The following Figure 6 conceptually represents two systems. Let us assume that these systems are actually organizations, perhaps to companies. Both organizations have complex point-to-point integrations within them. These integration links may represent actual systems integration or simply channels of communications between departments such as accounting, shipping and warehouse.



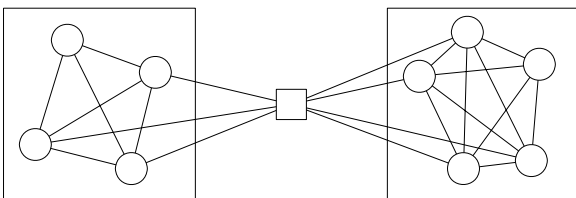
*Figure 6 Two organizations or complex systems*

To completely integrate the two organizations would look something like Figure 7. When two companies merge, they usually attempt some type of integration like this. Such integration, as we have seen, is unsustainable beyond two organizations.



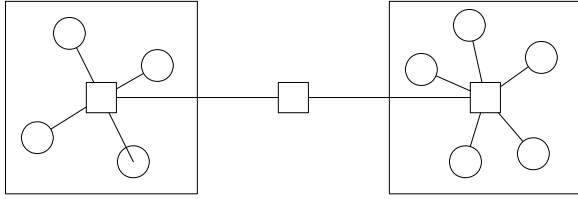
*Figure 7 Direct integration of two organizations or complex systems*

Thus, most companies implement an interface for integrating with other outside organizations, as shown in Figure 8. EDI is an example of such an interface. The web is also an example. The website provides a single point of integration with outside organizations, even if it is read-only.



*Figure 8 Indirect integration of two organizations or complex systems*

If the standard interface is exceptionally good, then organizations come to realize that what works externally should work internally as well, as shown in Figure 9. We saw this realization with the adoption of intranets in the late 1990s. Companies brought the websites inside the company to communicate between divisions in exactly the same manner that they communicate with customers.



*Figure 9 Complete indirect integration of two organizations or complex systems*

The same sort of integration occurs with applications as it does with organizations. The two boxes in the figures above could represent large software systems. Applications provide application programming interfaces (API) to handle Figure 8. Simple Object Access Protocol (SOAP) is a universal API for use across the Internet. Some programmers have come to realize that they can use SOAP as a means to integrate the internal components of their applications and