

Lezione 9 modulo 3

In questo modulo affronteremo l'ultimo layer di Archimate, almeno sulla parte Core, che è quella che ci interessa e affronteremo quindi il Technology layer, Technology layer che quindi si aggiunge al Business layer e all'application layer visti in alcuni dei moduli precedenti. Così come per gli altri moduli, viene citato il capitolo all'interno della specifica Archimate, che fa riferimento, appunto, a tutti gli aspetti legati al Technology layer; l'URL è riportato in questa slide. Così anche per gli altri layer, anche per il Technology layer sono presenti tre aspetti principali: quello attivo, quello comportamentale e quello passivo. Riguardo agli elementi attivi, si distinguono all'interno del Technology layer il concetto di device, che identifica una risorsa fisica IT, su cui è possibile eseguire del software, o su cui è possibile trovare degli artefatti e poi daremo una definizione, appunto di artefatto quale elemento passivo, all'interno di questo livello. Poi abbiamo il System Software, il software di sistema, che è un software che fornisce delle operazioni, delle funzionalità di base, che sono utili poi per costruire le applicazioni che ci serviranno. Abbiamo un altro elemento che è il Communication Network, attraverso il quale noi possiamo andare a definire, in maniera più o meno dettagliata, quella che è la struttura di comunicazione grazie alla quale è possibile mettere in interazione due o più dispositivi. E infine, abbiamo il concetto di Path, che può essere visto come un po' un'astrazione della Communication Network, attraverso il Path io identifico unicamente il fatto che è possibile identificare una connessione fra più nodi senza entrare però nel dettaglio tecnologico, che mi permette appunto di realizzare questa comunicazione, dal punto di vista infrastrutturale, ma solo dal punto di vista funzionale. Per fare un esempio, qui vediamo come gli elementi attivi possono essere composti. In questo caso, quello che si vede, è che abbiamo un device, che è stato chiamato Desktop PC, su cui è stato installato un sistema operativo, in questo caso Windows, su cui è funzionante un DBMS server e un Text Editor. Notate come la relazione che viene definita, per identificare appunto la gestione di un applicativo da parte di un sistema o di un altro applicativo, fa riferimento alla relazioni di tipo assignment; quindi, il sistema operativo è installato e quindi in gestione all'elemento fisico, appunto il device e il sistema operativo ha in gestione questi due applicativi. Questi due applicativi sono definiti come dei sistemi software, dei software di base, dei software di sistema, non sono specifici per le nostre applicazioni, possono essere usati poi per le nostre applicazioni, ma non sono elementi specifici delle nostre applicazioni, ma sono di uso comune. Questo è un po' il metodo di giudizio a grana grossa che possiamo utilizzare, per distinguere quelli che sono gli elementi applicativi, il software visto a livello applicativo, già visto anche a livello applicativo col concetto di modulo, da quello che invece è un software di sistema. Poi vedremo anche qualche altro esempio che ci aiuta per questa distinzione. In questo caso, quello che è rappresentato è che esiste un altro PC, in questo caso è un Laptop PC, è comunque un device, su cui è installato Linux e su cui è installato un web browser. Indipendentemente poi da quello che sto rappresentando, è giusto un esempio, io sto dicendo che questi due elementi sono collegati tra di loro, quindi è possibile definire un collegamento funzionale fra questi due dispositivi, grazie al quale è possibile mettere anche in comunicazione quelle che sono le applicazioni messe a disposizioni su questi dispositivi. Riguardo agli elementi attivi, vi sono altri elementi attivi che possono essere utilizzati: qui sono indicati tre, anche se in realtà utilizzeremo principalmente il primo, il nodo. Il nodo non è altro che, potremmo dire, una risorsa generale, che può essere anche composizione di altre risorse, che possono essere fisiche o meno, che sono rilevanti all'interno della nostra infrastruttura IT che vogliamo modellare con il livello tecnologico. Poi, facendo riferimento agli stessi elementi degli altri livelli in Archimate, quindi livello applicativo e livello di business, esiste anche il concetto di Technology collaboration, come un'aggregazione di due e più nodi che insieme cercano di raggiungere un determinato obiettivo e il concetto di Technology interface, che è il punto di accesso che mi permette, appunto, di utilizzare quelle che sono le funzionalità messe a disposizione di un nodo; anche se possono essere in qualche contesto abbastanza utili, per semplicità il Technology collaboration e il Technology interface, soprattutto, non verranno considerati nei nostri esempi, se non eventualmente in minima parte. Vediamo ora un esempio di utilizzo degli elementi attivi. In questo esempio vediamo come, utilizzando i nodi, abbiamo raggruppato i due device, che sono stati definiti prima,

come gli elementi su cui venivano installati i software di base. Come si può notare, in questo esempio, vediamo che i due nodi, che sono dei raggruppamenti logici, sono in qualche modo collegati tra di loro, attraverso un percorso. Ecco, il percorso è solo un'astrazione, che definisce la possibilità di collegare questi due nodi, senza entrare nel dettaglio tecnologico di come questo avviene, se non indicando eventualmente il protocollo. Questo vuol dire che si astrae da quella che è l'infrastruttura di telecomunicazioni, che permette questo tipo di connessione. È anche possibile, utilizzando invece il networking Communication, quindi un elemento diverso di tipo attivo, andare a definire eventualmente quelli che sono i dispositivi di rete che rendono possibile questa comunicazione. In questo caso, noi andiamo a definire un nodo, che si chiama, generalmente, in questo caso, Office 1 server, che contiene appunto un dispositivo, su cui poi viene installato anche il DBMS server e questo lo lega al fatto che viene considerato, all'interno della mia organizzazione, come un server. In questo caso, abbiamo anche tenuto a sottolineare che su questo dispositivo c'è anche un Text Editor; ovviamente, all'interno di un dispositivo, ci possono essere tantissimi applicativi, tantissimi software di base. Ecco, andranno ovviamente modellati solo quelli che possono essere considerati rilevanti ai fini del modello, del diagramma e dell'architettura che stiamo definendo in quel determinato momento. Dall'altra parte, abbiamo un nodo generico, questo che è chiamato Node, in cui abbiamo un PC laptop, anche qui non è che entriamo nel dettaglio di quelle che sono le caratteristiche, ovvio, se è necessario farlo, si può farlo utilizzando le proprietà di Archimate, senza andare necessariamente a scrivere tutti i dati all'interno del label dell'oggetto qui sul diagramma, su cui è importante, magari in questo caso, sottolineare il fatto che c'è un sistema operativo diverso da quello che offre il server, l'unico elemento di sistema installato e rilevante è quello del web browser. Passando agli elementi comportamentali, anche qui abbiamo il concetto di processo, il concetto di funzione, il concetto di Technology interaction, il concetto di evento, che fanno riferimento più o meno alla stessa semantica che abbiamo visto per gli altri livelli. Ci concentreremo principalmente sul Technology Service, il Technology Service che è appunto un servizio, a livello tecnologico, che può essere utilizzato o da altri elementi a livello tecnologico, oppure è quell'elemento che noi sappiamo benissimo fare da ponte fra un livello e un altro livello, nello specifico, fra il nostro livello tecnologico e il livello applicativo. Come potete intuire, quando adesso andiamo a introdurre il livello tecnologico, vedere le caratteristiche del livello tecnologico, andremo ad approfondire solo una porzione degli aspetti, perché, dal punto di vista IT, l'infrastruttura può essere abbastanza complessa e quindi ci potrebbero essere tantissime sfumature che non ha senso andare ad approfondire in un corso introduttivo, in un modulo introduttivo di Archimate. Quindi in questo caso, innanzitutto non avremo, come negli altri livelli, quello applicativo e quello di business, un pattern di base, ma lavoreremo su esempi tipici di deployment e di infrastrutture IT, che potremo utilizzare come punti di riferimento poi per specificare la nostra architettura. Questo, legandolo anche al concetto di applicazioni, a tier e la divisione di un'applicazione a layer, visto anche nelle lezioni di questo corso. Sempre facendo riferimento agli elementi comportamentali, anche se non andremo a approfondire questo aspetto, questo è un esempio che si potrebbe trovare all'interno del nostro livello tecnologico: per esempio, in questo caso, l'evento è il primo giorno della settimana, fa scatenare perché questo è pur sempre un trigger, così come per gli altri livelli, una funzione, o meglio un processo tecnologico, che non fa altro che eseguire l'incremental backup. Ecco, vedete, qui non ha senso metterlo a livello applicativo, non è un'applicazione, perché non è specifico per quello che è il servizio che voglio offrire a livello di business, ma è un servizio tecnologico che funziona dietro le quinte e che caratterizza la mia struttura IT. Quindi, anche se potrebbe essere visto come un software, in realtà è messo unicamente, è nascosto, potremmo dire, unicamente a livello IT. Poi c'è l'attività di verifica che il backup sia andato a buon fine e quindi c'è la possibilità, o di ritrovarsi con l'evento backup succeeded, piuttosto che failed, nel caso in cui il backup non è andato a buon fine. Passando quindi agli elementi passivi, ci concentreremo su un unico elemento passivo, che è quello dell'artefatto, che è un pezzo di dato, che potremmo in qualche modo associare al concetto di file, che può essere prodotto e utilizzato all'interno della mia infrastruttura IT. Qui vediamo un esempio che ci aiuta anche a capire un po' meglio la differenza fra un sistema software, potremmo chiamarlo anche un sistema di base, un software di base, meglio, e un'applicazione. Io qui sto rappresentando il fatto che ho un PC



desktop, su cui è installato un sistema operativo e il sistema operativo mi offre un servizio, un servizio tecnologico che è la possibilità di eseguire un'applicazione. E questa applicazione non è altro che l'applicazione che è memorizzata all'interno di un file eseguibile, che in questo caso rappresenta il file eseguibile del mio CRM, rappresentato appunto come un artefatto. Io cosa sto dicendo qui? Sto dicendo che grazie al mio sistema operativo, quello che è un programma, può diventare un processo. Partendo dal presupposto che sia chiara la distinzione fra programma, quindi un'entità statica, memorizzata all'interno di una memoria di massa e processo, quindi un'entità dinamica, che vive all'interno della memoria centrale e quindi può essere effettivamente eseguita, io qui sto dicendo che grazie a questo servizio, che probabilmente sarà implementato dal gestore dei processi del mio sistema operativo, è possibile istanziare questo artefatto, metterlo in memoria e dare la possibilità a questo elemento passivo, di essere, potremmo dire, un elemento attivo. Quindi in questo caso, io riesco in qualche modo a dire <<c'è un'applicazione, che è possibile mettere a disposizione grazie all'esecuzione di di questo artefatto.>> La differenza fra sistema software applicativo diventa più evidente nel momento in cui vado a connettere il livello tecnologico con il livello applicativo. Anche se non utilizzeremo la classica struttura, così come abbiamo visto per esempio fra livello applicativo e livello di business, abbiamo pur sempre una distinzione netta fra quello che è il livello applicativo, che è fornito in questo caso dalla parte bassa del livello applicativo, la parte, diciamo così, che abbiamo chiamato non visibile, composta dall'elemento funzione e dall'elemento modulo, e la parte, invece, più visibile dell'elemento, invece, tecnologico, del livello tecnologico, composto da un servizio e non tanto, qui lo nasconderemo, dall'interfaccia, ma dagli artefatti. Verrà molto più semplice poi capire, attraverso questo, la connessione che avviene fra questi due livelli. Nello specifico, cosa sto dicendo? Sto dicendo che, grazie al fatto che il servizio di esecuzione di un'applicazione, fornito da un sistema operativo, installato su un determinato PC, io posso prendere questo programma e dargli la possibilità di offrire le sue funzionalità veramente, quindi di diventare quel processo, inteso alla 'sistemi operativi', non il business process, quel processo che non fa altro che offrire le funzionalità, che corrispondono alle funzionalità messe a posizione dal modulo. Il modulo non è altro che quell'oggetto logico, quel componente software di tipo logico, che è realizzato appunto dal mio artefatto. In questo modo io riesco a rendere evidente il fatto che alcuni elementi sono più elementi applicativi, il mio CRM sono caratteristici del mio portafoglio applicativo e altri software, sebbene sono software così come quelli che abbiamo a livello applicativo, sono software di sistema, quindi, diciamo, hanno un'importanza meno rilevante rispetto a quelle che sono le applicazioni che effettivamente vanno a costituire gli elementi su cui poi poggerà il mio processo e anche il servizio che voglio offrire ai miei utenti finali. Vediamo ora, attraverso alcuni esempi, com'è possibile utilizzare il Technology layer, in combinazione anche con l'application layer, per andare a descrivere le modalità di deployment di una determinata applicazione. Come ho detto in precedenza, va distinto quelli che sono i software di base dai software applicativi. Infatti, i software applicativi vengono collegati direttamente a livello applicativo del nostro diagramma a livello appunto superiore, mentre i software di base, sono in qualche modo definiti solo all'interno della parte più nascosta del livello tecnologico. In questo caso, io vado a raccontare quella che è la una tipica One-tiered application. One-tiered application che però è divisa in due layer, perché vado a distinguere quella che è una porzione inserita all'interno di un file eseguibile, che contiene tutta la logica applicativa e la logica di presentazione, distinta appunto da quella che è la logica di gestione dei dati, che viene affidata invece a una porzione gestita direttamente da un DBMS, secondo quella struttura che abbiamo visto anche in precedenza. In questo caso, noi abbiamo un DBMS che fornisce questo servizio di accesso a un database, nello specifico il database su cui si fa riferimento è quello del CRM, che realizza quel data object che è stato definito a livello applicativo. Qui il desktop PC è, essendo appunto One-tiered, l'unico elemento fisico su cui si innesta sia il database, sia il mezzo con cui accedo al database, quindi il DBMS, sia l'ambiente che mi permette di eseguire, attraverso il servizio offerto dal sistema operativo, di eseguire il mio programma, che contiene appunto il modulo del CRM, quindi che istanzia appunto il modulo del CRM. La funzionalità quindi offerta grazie a questo modulo, è una funzionalità che sfrutta questi due elementi e questi due elementi, questi due servizi, in questo caso non fanno altro che andare a corrispondere ai due layer applicativi, intesi come layer logici, così come è

stato visto nella distinzione fra layer e tier, che sono caratterizzanti rispetto a questo esempio specifico. Questo tipo di soluzione di tipo One-tiered diventa ancora più chiaro, nel momento in cui vado ad introdurre il nodo. In questo caso, anche visivamente, non cambia nulla dal punto di vista della struttura, ma anche visivamente, io non faccio altro che andare a sottolineare il fatto che ho un unico nodo, su cui si innestano tutti gli elementi e, appunto, questo unico nodo offre un'applicazione che è stata divisa in due layer logici, quindi un layer fisico, a livello di nodo, due layer logici a livello applicativo. Passiamo quindi a un esempio di tipo Two-tiered. In questo caso, dal punto di vista più alto rispetto al nostro modello, non cambia nulla, quello che abbiamo rappresentato qui rispetto all'esempio precedente, perché abbiamo sempre il servizio di accesso al database che mi permette di offrire il database, il servizio di esecuzione dell'applicazione che mi permette di offrire il processo associato a questo programma, la cosa che cambia è il modo in cui questa applicazione, questi due layer logici, vengono suddivisi all'interno di due tier fisici. I due tier fisici vengono sottolineati dalla presenza di questi due nodi, uno l'abbiamo chiamato CRM DB node, l'altro Desktop client node, vado a dire che il mio database, che è sempre installato su una versione di Windows, in questo caso abbastanza datata, Windows NT, viene offerto attraverso sempre Microsoft Access, però dal punto di vista del layer applicativo non cambia assolutamente nulla. Dall'altra parte, noi abbiamo invece un'altra macchina, su cui c'è un Desktop PC e ipotizzo di avere una macchina Windows, una macchina Windows senza specificare la natura, la versione del sistema operativo, su cui invece viene a funzionare la parte di logica applicativa e logica di presentazione della mia applicazione. In questo caso sottolineo attraverso il path la presenza dell'utilizzo di un protocollo particolare, che è quello di DBC, di accesso, appunto di connettività verso dei sistemi DBMS, che mi permette la comunicazione fra questi due nodi. In questo caso non vado a sottolineare come è strutturata la mia rete, ma semplicemente creo un'associazione fra questi due nodi, andando a indicare però qual è il protocollo applicativo che vado a utilizzare per questo tipo di connessione. Andando a concludere, vediamo cos'è l'applicazione di tipo Three-tiered: in questo caso, sempre la mia applicazione, adesso sono state cambiate alcune tecnologie e dopo le approfondiremo anche nel prossimo modulo, dove vedremo degli esercizi aggiuntivi, qui cosa abbiamo? Abbiamo un'applicazione a tre livelli, dove ho sempre il lato server, dove (server inteso come DB) la logica applicativa in questo caso è demandata a un web server, ed è staccata rispetto a quella che è la logica di presentazione, che viene gestita dal mio client, all'interno del mio client e, in particolare, cosa sto dicendo? Sto dicendo che il mio database funziona su una macchina dedicata, detta DB server, su un nodo, quindi dedicato, che contiene questo device, su cui è installata una distribuzione Linux e su cui è installato il software di sistema, in questo caso il DBMS scelto non è più Access, come prima Ma è MySQL. Questo ha in carico anche la gestione della memorizzazione del mio database e grazie al DBMS, notate che la freccia di realize parte dal software system, non parte dal nodo, grazie al mio DBMS io ho la possibilità di accedere appunto a questo database. Dall'altro lato, la logica applicativa prevede che esista un device dedicato al CRM server, su cui ho un sistema operativo ben definito, con la sua distribuzione e, in questo caso, utilizzo un application server che è Tomcat, che mi funziona e che mi offre un servizio che è il servizio di esecuzione della web app. Perché sapete benissimo che in questo caso il ciclo di vita della mia applicazione web, non è gestito direttamente dal sistema operativo, ma è gestito dall'application server. Infatti non ho più l'executable file come prima, ma ho il distribution file, nel caso di Tomcat è un ".WAR", che viene gestito da Tomcat e viene messo in esecuzione grazie alle funzionalità offerte da Tomcat. Infine, il client node, che è il nodo su cui funziona la logica di presentazione, richiede un Desktop PC su cui adesso in questo caso viene richiesta la presenza di Windows, viene indicata la presenza di Windows, adesso stiamo facendo giusto un esempio e su cui funziona un web browser, che è quell'elemento che appunto mi serve per accedere alla mia applicazione, considerato un software di sistema. Che mi offre la funzionalità di web browsing. In questo modo io cosa sto dicendo? Sto dicendo che, chi utilizza questa funzione a livello applicativo, offerta dal CRM, la può ottenere grazie alla combinazione di questi tre aspetti principali, di questi tre servizi, è un'applicazione di tipo web che ha bisogno di un application server e di un DB server. E in questo caso, questi tre livelli logici che sono rappresentati, si può dire, da queste tre frecce, sono deployati in tre livelli fisici e questi sono identificati dai nodi.

