**VIDEO 9.30 da 6.40**

Salve di nuovo, io sono Andrea Abbondanza, sono laureato in scienze informatiche e mi sono occupato appunto dello sviluppo tecnologico di quello che è il progetto ReAsset. L'ingegner Ferone vi ha illustrato il perché, ora io voglio illustrarvi il come, quindi come abbiamo realizzato quel sogno che era il progetto ReAsset. Partiamo subito da una cosa che può sembrare banale, ma non lo è. ReAsset lo si può vedere come un insieme di componenti tremendamente eterogeni tra loro. Noi qui parliamo di blockchain, di IoT, di app per cellulari, software, web, software per PC, sensoristica, insomma capite che sono tanti aspetti tecnologici che tra loro hanno modalità diverse di funzionamento. Quindi ReAsset lo si può vedere anche come un insieme di tanti piccoli progetti e questi progetti tra loro devono comunicare. Partiamo dall'aspetto più dirimente di ReAsset, la blockchain. Da quel che ho capito voi avete scelto il progetto anche per la presenza di questo componente. Io per essere sicuro di collegarmi bene a quello che è stato precedentemente detto anche dall'ingegnere, voglio che sia chiaro, e se già lo è fermatemi che vado avanti, che voi sappiate cos'è una blockchain e sostanzialmente come funziona, non da un punto di vista tecnico approfondito, quantomeno da un punto di vista logico funzionale. Partiamo dalla cosa più banale. La cosa più banale è che una blockchain, a differenza di altri tipi di database o di registri, non è centralizzata. La blockchain è un ibrido tra rete decentralizzata e distribuita. Partiamo da questo piccolo presupposto, una rete centralizzata. Voi immaginate di dover fare una transazione economica, un bonifico.

Il vostro bonifico passa attraverso la banca, viene autorizzato dalla banca e poi va all'altra persona. In una rete decentralizzata invece non c'è un'entità che decide, ma ci stanno una serie di nodi che possono approvare o meno una transazione mediante una cosa chiamata meccanismo di consenso. Una rete distribuita invece è praticamente una rete dove in ogni nodo di questa rete è presente l'approvazione di meccanismo.

**Video 22.30**

Bitcoin non è che ogni nodo della rete può effettuare approvazioni, il famoso mining per approvare una transazione, ma ci sono anche nodi client che semplicemente si affacciano sulla rete e verificano lo stato. Non so se è chiara la differenza quindi tra centralizzata, scusate, decentralizzata e distribuita. La distribuita ogni nodo ha equal potere e all'intera copia del registro, quella centralizzata no. Ci sono dei nodi principali che hanno questo potere di poter approvare o meno le transazioni e poi ci sono dei nodi secondari. Perfetto, una delle più importanti proprietà della blockchain è quella di garantire l'immutabilità dei dati e questo poi comporta anche la possibilità di tracciare i dati perché se io so che un dato non può essere modificato

sono sicuro che il suo tracciamento è affidabile. In questa slide spiego un attimo il meccanismo di come avviene questa cosa. Sostanzialmente, si parte da un concetto, una funzione di hash. Una funzione di hash è una funzione molto semplice, matematica, che ovviamente poi può essere resa complessa quanto si vuole, ma che deve avere praticamente una proprietà, quella di essere irreversibile. Nell'istante in cui io effetto l'hash di qualcosa, mi deve comparire un dato da cui non si può risalire alla sorgente. Voi immaginate le vostre password, quando entrate nella casella mail, sul database del vostro gestore della casella non sta memorizzata la vostra password. Se avete password Andrea, non sta memorizzato Andrea, ma sta memorizzato l'hash che è una stringa di solito di circa, parte almeno da 64 caratteri. Perché? Perché nell'istante in cui quel database per qualche ragione viene rubato, nessuno può risalire alla vostra password. E quindi si risolvono una marea di problemi. Una persona che ha più password per vari account potrebbe farsi afferrare tutto. In questo modo invece non è così. Questo meccanismo è stato applicato al concetto di memorizzazione dato. E come funziona? Quando io registro una transazione o n transazioni su blockchain, creo un blocco di dato. D'accordo? Questo blocco di dato contiene una cosa importantissima. Contiene l'hash della firma della raccolta dati del blocco precedente. Che cosa vuol dire? Vuol dire che nell'istante in cui io dovessi andare sulla blockchain e modificare lo storico dei dati, quindi andare a cambiare il blocco precedente, la funzione di hash mi tornerebbe un altro valore. Quindi il blocco successivo saprebbe che il blocco precedente è stato modificato. E da qui l'immutabilità alla tracciata.

La funzione di hashing è una funzione che è usata da tanto tempo, da molti anni, e in altri modi. Per essere più chiari, la funzione di hashing deve essere deterministica, quindi sostanzialmente se dò una stringa ad Andrea, l'output deve essere sempre lo stesso, non cambia ogni volta. Deve essere ovviamente di rapida computazione, se devo fare un login e mando la mia password, non posso aspettare 5 minuti che lui capisca che sono io. E poi ci sta l'irreversibilità, che è quella che vi ho accennato prima, e quest'altra cosa si chiama effetto valanga. Sostanzialmente, immaginate una cosa del genere. Andrea viene tradotto in una stringa, ora prendiamo la stringa Fabrizio, non può essere tradotto nella stessa stringa. Matematicamente questa cosa non è garantita, però una buona funzione di hashing ha un basso livello di collisioni, che vuol dire che sostanzialmente è difficile che queste stringhe finiscano allo stesso modo. Ecco la digressione, non so se avete mai sentito parlare del protocollo hashing MD5, questo protocollo è stato usato fino agli inizi degli anni 2000, adesso è stato tolto da mezzo perché sostanzialmente, essendo la stringa di output molto corta, gli hacker hanno iniziato a memorizzarsi tutte le stringhe di partenza, e quindi se si trovano la stringa BBBB, sanno che magari corrispondano Andrea, Fabrizio e Giuseppe. E quindi si è passato a sistemi più avanzati, SHA, 128, 253, 512, come quello della blockchain. La blockchain nel nostro caso usa una funzione di hashing SHA 512. Ah, ho giusto un piccolo esempio, se leggete sotto è proprio un esempio di MD5, la stringa hello world diventa quella roba lì che vedete.

Se voi vedeste una stringa di hash di un SH512 è tipo 6 volte la lunghezza di quella lì. Perfetto, data l'introduzione di qualsiasi blockchain e come funziona, almeno da un punto di vista basilare, mi collego un attimo a quello che diceva prima l'ingegnere Ferone. Nell'istante in cui io creo una community, dove ci sono i vari attori relativi al detentore, appapltatore, i fornitori e così via, come posso fare in modo che queste persone lavorino al meglio tra loro? Posso farlo monitorando. Ma giustamente se io sono quello che monitora, si torna al problema della gestione centralizzata. Perché un detentore, un fornitore, un appaltatore deve fidarsi di me? E qui rientra in gioco la blockchain. La blockchain consente di creare con la sua trasparenza e mutabilità dei dati e soprattutto con la sua proprietà di identificazione, perché una blockchain necessita che tutti gli attori siano identificati, di creare un ambiente trustless. Trustless vuol dire un ambiente senza fiducia. Che vuol dire? Dato che io ho dei soggetti che non si fideranno mai tra loro, allora devo togliere da mezzo proprio il concetto di fiducia. Io ti metto in un ambiente dove non hai necessità di fidarti perché il sistema che c'è sotto è trasparente e mutabile. C'è un esempio, un esempio pratico. Il manutentore che è identificato effettua l'operazione di manutenzione. Questo si collega a quello che diceva anche l'ingegnere prima. Nell'istante in cui io effetto la manutenzione, il mio sistema va a tracciare le performance dell'asset e valuta anche i tempi che ci ho messo per ripararlo.

Dato che questi dati, come abbiamo detto, sono analizzati su blockchain, sono certi, sono immutabili, quindi se il detentore, che l'appaltatore, che il fornitore sa che sono veri e sono accessibili, che cosa posso fare io detentore, in questo caso specifico? Io posso effettuare il calcolo delle performance nella operazione manutentiva ed eventualmente assegnare bonus se è stato particolarmente celere il fornitore e l'appaltatore, o un malus se invece magari su un asset monitorato che mi blocca la produzione è stato lento ad operare. E nessuno si può appellare a dire c'è un'opinione dietro a questo, perché sono dati oggettivi, immutabili e trasparenti, perché ogni attore della community ha accesso alla blockchain avendo un'identità. E i nodi essendo distribuiti, per essere chiari, qualsiasi degli attori di Reasset può avere un nodo di blockchain o dieci nodi di blockchain in casa sua, non sul cloud, non dal detentore, può averlo in casa sua, la trasparenza è garantita e quindi nessuno si può appellare. E già parliamo di quanti contenziosi possono scomparire in un contesto del genere. Ok, allora, ora si entra un pochino in un campo più tecnico informatico, quindi se dovessi scendere troppo in profondità avvisatemi che mi fermo. Come dicevo prima, Reasset è composto da tanti componenti diversi, che possono essere sia hardware, quindi schede, sensori e così via, che software. Nello specifico io ho individuato quelli che sono i macro moduli che compongono il progetto, sia lato software che base di dati, ossia ci sta un CMMS, che sostanzialmente è un sistema computerizzato per la manutenzione.

che in pratica non è nulla in meno del SIEM che diceva prima Lenin e Sperone, un modulo di analisi dati, un modulo di comunicazione con le UIP, le UIP sarebbe la nostra IoT, la vediamo tra un secondo, un modulo di comunicazione con le app esterne, e soprattutto il modulo di comunicazione con la blockchain, ma più di tutti la blockchain, la blockchain è un software. Basi di dati, le basi di dati sostanzialmente sono il nostro modo per memorizzare i dati in modo persistente, qualsiasi cosa voi utilizzate oggi, che sia la mail, che sia Facebook, ha delle basi di dati, queste basi di dati possono essere di vario tipo, nel nostro caso la nostra base di dati principale è la blockchain, ma abbiamo anche delle basi dati normali, quelle che chiamiamo off-chain data storage, per esempio, sono i file, voi non potete mettere dei file sulla blockchain, la blockchain per sua norma cresce in modo esponenziale, se iniziate a metterci i file sopra, non avrete un numero di macchine possibili da poter conservare la blockchain, infatti il meccanismo qual è? Io prendo il file, ci faccio l'hash, e lo memorizzo su blockchain, se il file viene alterato io me ne accordo. E poi abbiamo sostanzialmente la componente del CMMS, che è quella là che diceva proprio l'ingegner Ferrone quando ha detto che esistono dei software che aiutano da un punto di vista documentale a fare la gestione e la valutazione, non è nient'altro che quello. A sinistra potete vedere quindi una cosa che si chiama schema UML, che è un linguaggio informatico sostanzialmente, se mi fate passare il termine, che rappresenta queste componenti e come comunicano tra loro. Abbiamo in basso a destra quella che è la blockchain, in alto a destra il CMMS, a sinistra invece abbiamo tutto quello che sono i moduli di Reasset, quindi modulo di intelligenza artificiale, modulo di comunicazione app e così via. In basso a sinistra ci sono le basi dati convenzionali, quindi file, database classici, relazionali e via. Perfetto, questa forte eterogeneità però come la possiamo mettere insieme?

In informatica, negli ultimi 10-15 anni, è nato il concetto di micro servizio. In realtà, anche da prima, però, sostanzialmente, in informatica dovete capire che c'è l'idea, l'hardware non riesce a supportare questa idea, l'idea viene accantonata. Dopo 10 anni, l'hardware diventa più potente e l'idea ritorna. Le macchine virtuali che oggi si usano quotidianamente stavano già negli anni 80, ma non c'erano macchine così performanti da farle girare in modo decente. Il micro servizio, che cos'è in pratica? Il micro servizio potete vederlo come un modulo software che fa una cosa. Io sono il modulo software, per esempio, dell'accesso utente. Io mi occupo esclusivamente di quella cosa. Pensate ad Amazon. Amazon ha il modulo accesso utente, ha il modulo del carrello, ha il modulo degli ordini acquistati, ha un modulo della vetrina. Questi moduli comunicano tra loro. Perché questo sistema negli ultimi anni è diventato così importante? Perché negli ultimi anni, con l'avvento del web massivo, che è quello che noi utilizziamo quotidianamente, che può essere Facebook, Amazon o qualsiasi altra cosa, immaginate un sistema monolitico, così si chiamava, di sviluppo software precedente. Io creo un programma, un pezzo di questo programma crasha, si blocca tutto. Non funziona più niente. Ora immaginate che Amazon, dove crasha il modulo dei preferiti, il resto continua a funzionare, io posso continuare a fare gli acquisti, accedo, compro, mi arriva l'ordine, non vedo i preferiti. Questo è il primo vantaggio del micro servizio. Oltre che poi, dato che Reasset è un tipo di progetto che utilizza la blockchain, quindi immaginate che non gira su un unico server. Già di base sua la blockchain, per essere distribuita, deve stare su tanti server, altrimenti non ha molto senso. È un software normale se ci metto tutti i nodi su un unico server. Il micro servizio aiuta anche in questo, con la scala abilità. Perché nell'istante in cui io ho un e-commerce, ho 10 utenti, mi basta un computer da 8 giga di RAM, per esempio. Gli utenti diventano un milione, non è che posso mettere risorse infinite su una macchina, ma che cosa posso fare? Posso creare macchine parallele. Ci metto il micro servizio su ognuna e poi ci sta un oggetto.

che, ricevuto una richiesta, lo sposta su un micro servizio adibito quindi altro vantaggio informatico del micro servizio Quindi abbiamo il discorso della scalabilità, abbiamo il discorso che se c'è un fault non crasha tutto il sistema, e così via. Ma in aggiunta a questo, sostanzialmente non ci mettiamo i container I container per farvela molto semplice sono un'evoluzione delle macchine virtuali Le macchine virtuali che cosa fanno? Prendono il vostro computer, lanciano questo software e creano uno strato, sempre software dove questo strato fa finta di essere un computer, quindi c'ha delle funzioni che fanno il calcolo del processore e così via Ovviamente questo richiede tantissimi risorse, sono pesanti I container invece, qual è il vantaggio che hanno? I container si poggiano direttamente sulle risorse hardware, quindi sulla vostra scheda del vostro processore, ma sono comunque degli ambienti isolati. Perché ci sta questo vantaggio? Perché immaginate che io ho un modulo di intelligenza artificiale questo modulo di intelligenza artificiale è sviluppato in Python, perché Python è il linguaggio principale dell'intelligenza artificiale Poi ho un modulo web, che magari è sviluppato con il linguaggio Javascript e Node.js perché quello è uno dei più utilizzati per fare il server web. Se facessi un unico software su un'unica macchina avrei il problema di dover installare tutta questa roba in un unico contesto e mi ritorno il problema della pesantezza, il problema che se una di queste cose cade, cade tutto e così via. Invece io ho il mio container dentro ci metto le librerie, ci metto il codice e lo lancio e lui funziona come micro servizio. Prendo un altro container, ci metto il codice ci metto le librerie e lo lancio. Tutti questi oggetti tra loro comunicano questa comunicazione si chiama orchestrazione di micro servizi e creano una piattaforma. Questo è un esempio pratico di un micro servizio implementato in Reasset. Questo micro servizio praticamente ha un unico scopo, perché ovviamente un micro servizio, se fate un micro servizio che ha tanti scopi diversi, vi torno il problema che se cade una cosa, cade tutto il resto. Questo micro servizio ha il compito di comunicare con il nostro IoT quindi al suo interno ha un modulo che si chiama broker questo broker non fa

altro che prendere i messaggi dall'IoT e l'inoltra nell'altro moduletto che si chiama Data Converter perché noi abbiamo questi messaggi che sono binari, li dobbiamo convertire in dati per poi memorizzarli sul blockchain e sui database. Questo oggetto qui, nell'istante in cui dovesse andare giù, abbiamo solo il problema che l'IoT non ci manda date al momento tenendo presente che l'IoT ha la sua copia locale ma tutto il resto di Reasset continuano a funzionare quindi il detentore continua a vedere i suoi asset, l'appaltatore può continuare a fare schetulazione della programmazione, della manutenzione, e così via. Perfetto, siamo arrivati alle UIP. Perché non le chiamiamo IoT? Voi l'IoT avete idea più o meno di cosa siano? Sostanzialmente per IoT, Internet of Things, si intende qualsiasi cosa che raccolga dei dati e comunichi con un server per Reasset questo non bastava come definizione perché, ragionate su un concetto, nell'istante in cui io memorizzo un dato sul blockchain quello è immutabile, sta lì, nessuno lo tocca più ma io devo sapere che il dato è corretto perché se inizio a memorizzare dati sballati, apro immutabilmente dei dati sballati sul blockchain quindi le nostre UIP sono state sviluppate in che modo? Con tutta una serie prima di meccanismi di sicurezza come il TPM, qualcuno l'avrà sentito, per cui Windows adesso lo richiede per esempio come diceva l'ingegner Ferone con il discorso che l'operatore si deve identificare che può essere con un utente password, può essere con un bluetooth che si avvicina e viene identificato automaticamente ma io devo essere certo che quella persona che sto operando è la persona che deve operare io non posso mandare un idraulico a fare il lavoro di un elettricista per essere chiaro, in più le UIP montano a bordo dei sistemi di intelligenza artificiale mediante delle logiche fazzi, le logiche fazzi sono una delle tante modalità di intelligenza artificiale che si possono trovare effettuano delle valutazioni sull'andamento del componente prima l'ingegner Ferone ha detto una cosa importante, ha detto oggi tendenzialmente si va a guasto

Cioè, si guasta la macchina e io vado a ripararla. L'OIP invece che cosa fa? L'OIP, nell'istante in cui vede che la curva dell'andamento di prestazione, per esempio, c'è troppo assorbimento, la temperatura è troppo alta e così via, se ne accorge e manda degli alert in modo da consentire all'operatore di manutenzione di intervenire in modo preventivo, ma non preventivo a sensazione, tipo, vado lì e magari butto tutti i soldi perché quell'asset sarebbe andato avanti per un altro anno, ma preventivo, ossia prima che si blocca la produzione, si blocca il processo e l'asset si guasta, io già so che cosa devo andare a fare. A destra e sotto, vedete quella foto con tutti quei cavi, quello è stato il nostro impianto pilota dove abbiamo testato l'OIP. L'OIP, se vedete in pratica, è quella scheda in alto a sinistra più il monitor. Non si vede tanto però comunque dovete immaginarla come una piccola scheda madre con le antenne e c'ha il suo monitor barra tablet con cui interagire. A destra vedete lo schema effettivo. Lo schema effettivo è composto sostanzialmente da un pannello solare, un inverter, una batteria e un carico. Il carico sarebbe la luce, quindi quello che vada ad assorbire quello che è prodotto dal pannello. In alto, quei cosetti bianchi che vedete sono dei trasmettitori BLE, Bluetooth Low Energy. Li utilizzate sul cellulare, quindi sostanzialmente una tecnologia consolidata, infatti si può accedere a quelle letture anche da cellulare per farvi capire. Questo impianto ci ha consentito di acquisire i dati effettivi reali, dati real time, su un asset che è il pannello solare. Questo, forse non so quanto vi può essere utile, questo si chiama Sequence Diagram, è un UML, fa parte sempre dell'UML, che rappresenta quello che succede nell'istante in cui c'è un caso d'uso di manutenzione straordinario, ossia l' O.I.T. manda un segnale qui.

O.I.T. manda un segnale e dice guarda questo oggetto si sta rompendo questo oggetto ha la temperatura nominale di 20°C ma sta a 40°C in realtà già a 25°C ve lo segnalo, però per darvi l'idea, quindi ci sono i vari attori che seguono tutto il filone allora a sinistra vedete gli attori quella schedina in alto a sinistra è la O.I.P. e invece i rettangoli che vedete sono il sistema Reasset quindi ci sta il sistema server, la blockchain e così via qui potete vedere l'interazione a sinistra invece vedete sostanzialmente un software che abbiamo sviluppato perché prima di avere le O.I.T. che ci tengo a sottolineare le O.I.P. sono state realizzate da zero non sono una cosa che comprate su Amazon attenzione usavamo un software per inviare delle segnalazioni e fare i test del CMMS che non si vede benissimo ma la schermata tra le due vedete ci sta un piccolo straccio di software quello non è nient'altro che una notifica del nostro CMMS che dice guarda il pezzo mi ha mandato un'anomalia quello che vedete in basso a destra è invece l'interfaccia dell'O.I.P. che indica sempre l'anomalia è in arancione, non si vede benissimo ma c'è scritto che c'è un'anomalia. Perfetto, andiamo quindi su un concetto importante facendo riferimento a quello che diceva in precedenza io ti devo pagare

**VIDEO 10.23 fino a 8.30**

Ma se non ho digitalizzato questi asset, su cosa programmo, cosa traccio? Quindi questo è un aspetto importantissimo. Ed è stato realizzato un software dedito esclusivamente alla digitalizzazione degli asset. Poi, per esempio, abbiamo il CMMS, che invece ci consente di poter effettuare il management di tutte quelle che sono le attività manutentive degli asset, di aggiornare, e così via. La schermata che vedete qui è del nostro CMMS, lato calendario, perché sono delle operazioni schedulate. Poi abbiamo l'Apple Core Center, ossia noi abbiamo un vantaggio, quello di avere una blockchain che ci registra tutte le identità. Nell'istante in cui, immaginatevi una scuola, io registro l'identità del personale ATA, quindi attenzione, qui non c'entra il manutentore, qui stiamo nella sfera del detentore. Il personale ATA è registrato, gli si dà l'applicazione, il personale ATA vede che ci sta una luce fulminata, vede un comvettore che fa rumore, effettua la segnalazione, tutto tracciato, tutto registrato. Quindi effettua la segnalazione, la segnalazione arriva all'avvalutatore, quindi al manutentore, e il manutentore deve proprio dire. Altrimenti il detentore non ne accorga se ti arrivano le segnalazioni e tu non le vai a schedulare, a sistemare. E quindi, diciamo, ci sta l'importanza di avere tutti questi altri software per completare il cerchio del processo che dà senso a tutto il progetto di Reasset. Allora, questo è uno schema che vuole racchiudere un po' tutto il processo di manutenzione e associargli i pezzi di Reasset che abbiamo sviluppato. Vedete, per esempio, in alto a sinistra ci sta una planimetria con sottoscritto IoT. Questo rappresenta la digitalizzazione, se io quando vado ad effettuare, la digitalizzazione ovviamente applico pure le UIP con la sensoristica. Nell'istante in cui ho il patrimonio digitalizzato, che cosa succede? Questo patrimonio mi finisce nel CMMS, d'accordo? Quindi vedete a destra io ho le schermate del CMMS. Ora, cosa è importante? Ogni operatore di manutenzione è un operatore di manutenzione e ogni contratto hanno dei processi che sono standard. Quindi quello che vedete a destra, al centro, è la definizione dei processi. In che senso? Nell'istante in cui ho necessità di effettuare un processo manutentivo, questo processo come funziona? Che io ricevo una segnalazione da un utente. La segnalazione va convalidata prima, magari da un manager manutentore. Questa segnalazione poi va incontrata ad un...

operatore di manutenzione, e alla fine deve essere inoltrata al detentore quando è stata conclusa, io ho gli strumenti per definire questo processo in modo che l'appaltatore faccia funzionare tutto come da contratto. E quindi c'è quest'altro modulo soft. Abbiamo a sinistra gli algoritmi di interpretazione esperta MQC, quelli di cui vi ho parlato pocoanzi, abbiamo le UIP, e poi c'è l'area di testing che è quella che vi ho fatto vedere prima. Questo schema parte dalla digitalizzazione, passa per la disegnalazione di errori, passa con la richiesta di eventuali diforniture, alla effettiva manutenzione con l'aggiornamento dell'anagrafica, e rappresenta lo schema standard di quella che è un'attività manutentiva. Ovviamente tenete presente che ognuno può declinare le fasi come meglio crede, ma la base è questa. Da questo punto di vista Reasset ci ha dato dei risultati. Il risultato concreto è che abbiamo realizzato quello che si chiama prototipo dimostrativo, cioè che possiamo qualcosa di funzionante, che esiste, sta lì, e con dei casi d'uso applicati in funzione. Quindi non è solo filosofia, ma noi abbiamo effettivamente del codice, di software. Abbiamo tutta una serie di componenti che vanno a creare una cosa che si chiama prototipo. Questo prototipo ci consente di validare l'implementazione di una comunità fiduciaria, che quella che vi ha introdotto prima l'ingegner Ferrone, di cui io prima vi ho parlato nell'ambiente trustless, ossia dei soggetti che stanno insieme, cooperano al meglio, perché non hanno necessità di doversi fidare l'uno dell'altro. La semplificazione e l'automazione dei processi, mediante i famosi smart contract che vi dicevo prima. Immaginate, faccio un esempio proprio banale. Un pezzo si sta rompendo, quindi la OIP ci manda la segnalazione. Questa segnalazione arriva al manutentore. Il manutentore oggi cosa dovrebbe fare? Dice, ok, devo sostituire la pompa, allora chiamo i fornitori, mi faccio fare le offerte, verifico i tempi, verifico quello che è il posto. Ricevo la pompa e la vado a sostituire. Immaginate questo processo invece gestito dallo smart contract, quindi tutto tracciato, tutto valido. Che cosa succede? Che la segnalazione arriva oltre che all'arma manutentore anche ai fornitori. I fornitori mettono subito a disposizione quello che hanno, uno smart contract vede le offerte dei fornitori e secondo i criteri che sono stati scelti nella configurazione sceglie quello che è il posto.

il pezzo con l'approvvigionamento e rendetevi conto che è saltata tutta una trafila burocratica che magari può impiegare giorni, a tempo di secondi. Monitorando il trattamento e realtà delle operazioni di manutenzione e quello di cui vi ho parlato prima, io effetto manutenzione posso essere anche valutato su quella manutenzione mediante una qualità di certezza e quello che si chiama sotto approccio win-win sostanzialmente cosa vuole dire? Se io creo questo ambiente dove io per esempio sono il fornitore che vantaggio ho a stare in Reasset? Ho il vantaggio che sono in mercato protetto perchè non devo fare concorrenza con ciascuno di voi o con quello che è amico dell'appaltatore per esempio ma su dati oggettivi, quindi io faccio un'offerta e se questa offerta va a rispecchiare quelle che sono le specifiche da contratto vado subito. Io detentore, qual è il mio vantaggio? Il mio vantaggio è che io posso finalmente pagare quello che effettivamente viene fatto e non pagare questo canone. Un altro vantaggio importante che ho è che io so esattamente qual è il mio patrimonio e qual è la sua consistenza. Immaginate un ospedale che ha migliaia e migliaia di asset ma tendenzialmente molti magari vengono dimenticati ma non dimenticati per mal di sfato perchè c'è troppa roba e io che sono il detentore non ho manco idea di quello che ho su tutto un certo aspetto. L'appartatore qual è il suo vantaggio? Il vantaggio è che si ritrova per esempio già la digitalizzazione fatta quindi non deve crearsi lui tutto il discorso diciamo dell'anagrafica tecnica nell'istante in cui il detentore fa l'anagrafica e ci sta un quadro elettrico per esempio se oggi sono appaltatore e prendo un appalto e vedo che il quadro elettrico non è a norma per me è un problema, perchè sono io il responsabile in quel momento ma nell'istante in cui questa roba è stata fatta prima io sono sicuro che quello che mi arriva è corretto e non mi devo aspettare sorprese e posso quantificare al meglio quella che è la mia offerta. Questo è un ambiente win-win Tutti quanti hanno un vantaggio nel stare in questo contesto la parte tecnica di Reasset ha consentito di validare questi concetti che sono poi alla base del progetto di per sé.