

ARCHITETTURA P-NET CLIENT-SERVER MULTI-TIER CON DATABASE

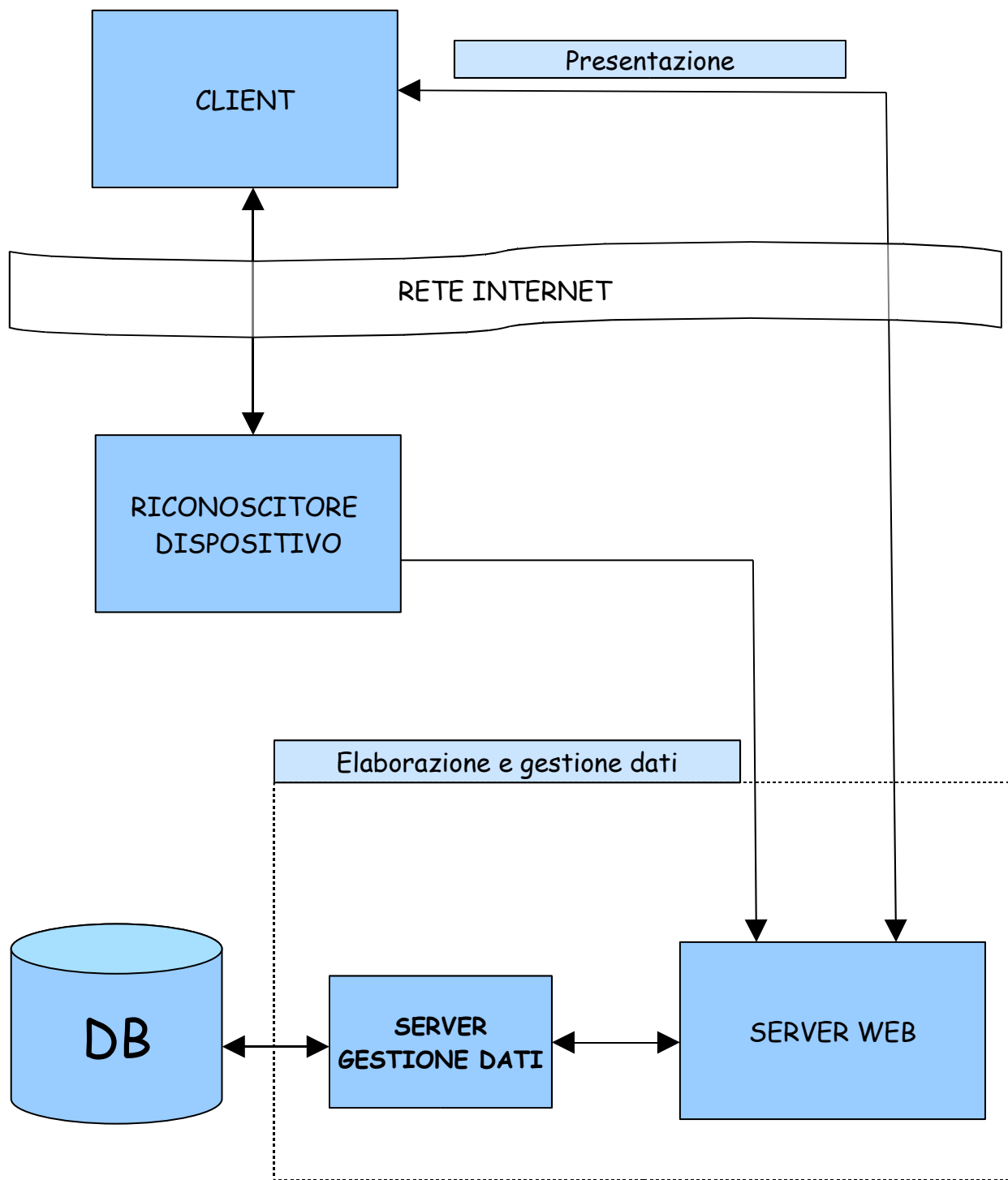


Fig. 1

L'architettura sopra riportata in figura 1 riflette la nostra idea di architettura di sistema e la direzione verso la quale ci siamo mossi. Tale schema ovviamente non rappresenta l'intero sistema ma ne copre una parte importante.

L'architettura da noi esposta è un sistema client-server che si basa sull'idea che l'elaborazione e la gestione dei dati siano svolte dal server e in generale dal sistema in modo che l'utente dunque risulti un thin-client e veda come una black-box il sistema. L'utente quindi tramite il suo browser web (e dunque la rete internet) installato su un dispositivo che, come abbiamo largamente analizzato, non necessariamente deve essere un pc, fa accesso al sistema, si logga e diventa un utente "pari" di P-Net. Il carico dunque è spostato tutto sul server web che si occupa sia del caricamento del sistema sul dispositivo utente, sia della comunicazione con il database. In merito a questo apriamo una piccola parentesi sul modello da noi prodotto: avendo pensato in aula all'intero sistema come un modello repository blackboard come gruppo abbiamo aggiunto un database relativo al sottosistema di nostra competenza utilizzato principalmente per memorizzare e consistere le varie versioni di P-Net da caricare sul dispositivo utente; ovviamente ciò non toglie che il db può essere usato anche per altri scopi come ad esempio mantenere le informazioni di login degli utenti (username e password) da controllare all'atto di ogni login dell'utente.

Ritornando all'architettura client-server da noi pensata è stato notato che il modello thin-client si adegua alle nostre necessità, ma presenta diverse inefficienze come il troppo carico di lavoro delegato al server che in principio doveva attuare il riconoscimento del dispositivo e provvedere al caricamento, senza contare che il server deve poter gestire più connessioni in un istante; tutto il carico del lavoro dunque sarebbe assegnato ad un solo processore. Si è dunque pensato di costruire una piccola e semplice architettura multi-tier ovvero a più livelli in cui ci sono più processori su cui è distribuito il carico. Si ha infatti un database e un server di gestione dati, esso li distribuisce al server web che si interfaccerà con il dispositivo utente e un modulo aggiuntivo a sé stante chiamato "riconoscitore dispositivo" che a sua volta si interfaccia sia con il server sia con il dispositivo; questo modulo si occupa del riconoscimento delle caratteristiche del dispositivo e del tipo di connessione utilizzata.

Il lavoro è distribuito, senza però far perdere di centralità al server web che provvede a gestire l'interazione fra le varie componenti non sobbarcandosi però tutto il lavoro.

Il server di gestione dati ha il solo compito di comunicare col database di effettuare le query e di fornire i dati richiesti al server web che li smisterà a seconda delle esigenze; nel nostro caso in particolare il server web comunica direttamente con il client.

L'architettura multi-tier presentata quindi permette l'ottimizzazione del trasferimento delle informazioni tra server web e server database tramite apposito protocollo (anche a basso livello) e la distribuzione dell'elaborazione.

Non bisogna dimenticare inoltre che il tutto è realizzato tramite la rete internet, cosicché la comunicazione è effettuata ad esempio tramite protocolli html, https, per

cui non può mancare l'infrastruttura di rete fondamentale.

Questa architettura dunque è atta a garantire efficienza e affidabilità (requisiti richiesti dal sistema) avendo distribuito il carico di lavoro e avendo delegato la gestione dei dati ad un modulo (server gestione dati) il cui sviluppo e la cui progettazione saranno orientate all'affidabilità e alla protezione dei dati con appositi meccanismi di recovery.

È garantita la portabilità del sistema inserendo un apposito modulo, ovvero il riconoscitore dispositivo, ed è garantita la possibilità di caricare diverse versioni di P-Net a seconda delle caratteristiche del dispositivo da cui si fa l'accesso; si assicura inoltre, ad esempio, il multitasking relativo al login utente, essendo il server web in gradi di ricevere più richieste contemporanee.

Con questa breve introduzione sono state presentate le scelte progettuali e architetture da noi effettuate e sono state motivate in relazione ai requisiti da noi precedentemente scritti.

INTERFACCE P-NET

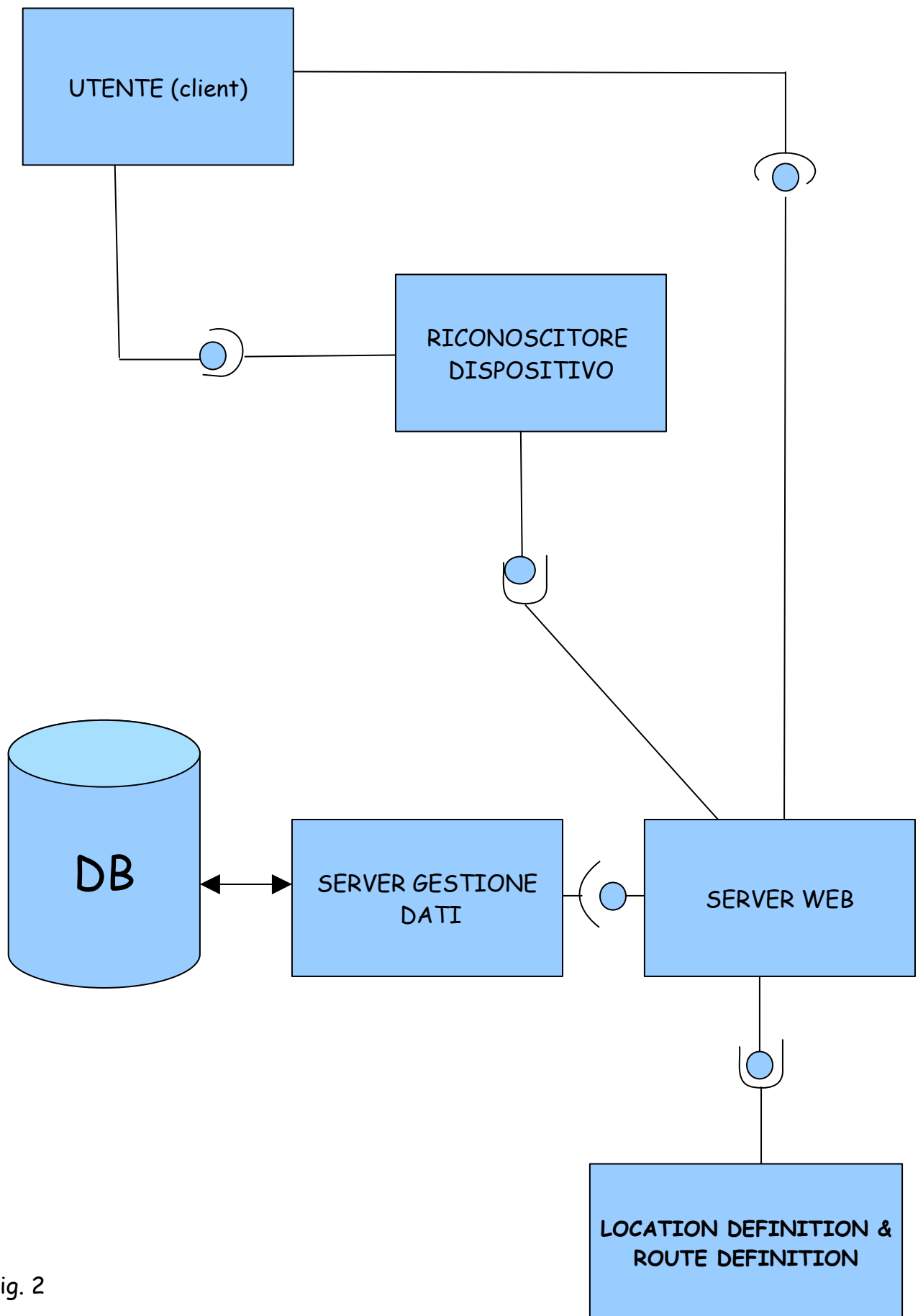


Fig. 2

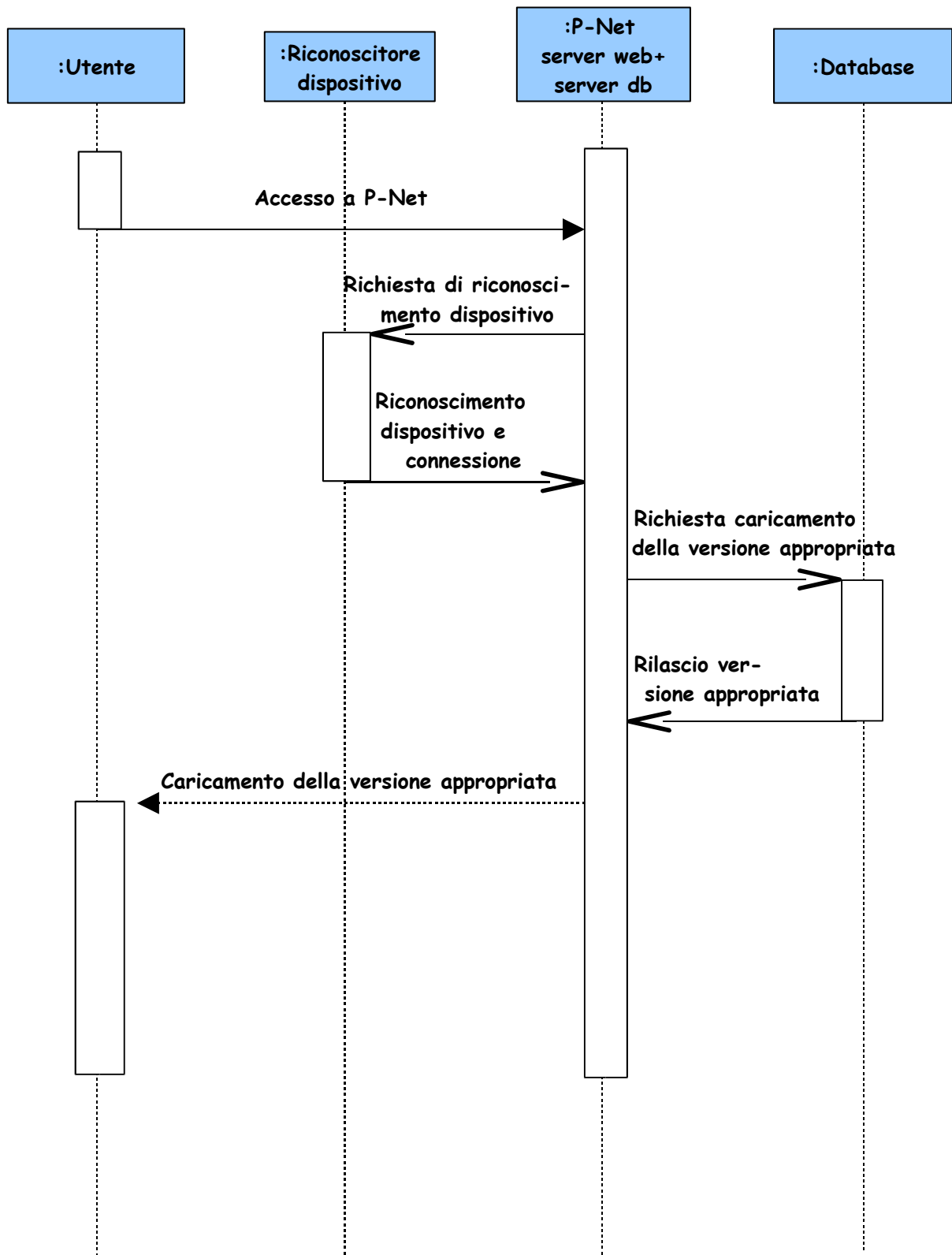
La struttura da noi definita in figura 2 è una struttura client-server che si rifà molto a quella di figura 1. Il client ovvero l'utente che fa accesso a P-net si interfacerà sia con il server del sistema sia con una componente di esso rappresentata dal sottosistema "riconoscitore dispositivo" atto a reperire le caratteristiche del dispositivo e il tipo di connessione da esso utilizzata.

Il server web di P-net è il fulcro dell'elaborazione, ma fa anche uso di un server gestione dati che gestisce il database da cui reperisce la versione del sistema (sito internet) da caricare sul dispositivo utente.

Abbiamo bisogno dunque di cinque interfacce una per la comunicazione diretta con il client da parte del server, una che permette la comunicazione con il client al riconoscitore dispositivo, una che fa comunicare il riconoscitore e il server web affinché esso sia istruito riguardo le caratteristiche del dispositivo e possa provvedere al caricamento della versione adatta di p-net e una tra il server gestione dati e il server web.

La quinta interfaccia fornita dal server ci permette di comunicare al sottosistema che si occupa della location e della route definition (le stesse caratteristiche da noi utilizzate per il caricamento della versione di p-net customizzata per l'utente).

DIAGRAMMA DELLA SEQUENZA DEL CARICAMENTO DELLA VERSIONE P-NET



Il seguente sequence diagram presenta sempre il concetto della portabilità su più dispositivi tramite riconoscimento caratteristiche e connessione, ma mette in risalto alcuni particolari importanti.

Innanzitutto notiamo la sequenzialità delle operazioni ovvero la loro scansione temporale e il tipo di comunicazione che può essere sincrona o asincrona.

Alla richiesta di accesso a P-net da parte dell'utente il server, nel nostro caso in particolare il server web, istruisce il modulo riconoscitore dispositivo richiedendogli dati relativi al dispositivo e alla connessione in modo asincrono; il fatto che la richiesta non sia "bloccante" per il server web è ovvio perchè esso deve gestire più login e quindi deve poter ricevere più richieste in uno stesso momento.

Ricevuti i dati dal riconoscitore dispositivo, che a sua volta manda un messaggio asincrono, il server db (istruito opportunamente dal server web) provvede a comunicare col database e a caricare la versione appropriata del sistema.

C'è notare infine che la comunicazione tra l'utente e il sistema è sincrona, infatti l'utente deve aspettare, oltre che siano riconosciuti e verificati username e password, che il sistema elabori i dati e provveda a caricare la versione adatta.