Abstract

Master di I livello in Analisi Dati per la Business Intelligence e Data Science

Studente: Michele Dallorto

Titolo della tesi: Dalla gestione *legacy-stateful* alla validazione *contract-driven*: transizione architetturale del software di acquisizione dati per esperimenti quantistici

La tesi analizza e documenta un intervento di refactoring su un sistema reale di acquisizione dati per esperimenti quantistici, sviluppato nel contesto del tirocinio presso INRiM e integrato con l'ecosistema Labber/Keysight QCS. Il problema di partenza non era l'assenza di funzionalità, ma la fragilità strutturale tipica di molti sistemi nati in modo incrementale: forte dipendenza dallo stato interno, accoppiamento elevato tra componenti, contratti impliciti tra client e server, bassa tracciabilità dell'esecuzione e difficoltà di evoluzione senza regressioni.

L'obiettivo del lavoro è stato quindi evolvere l'architettura senza interrompere l'operatività del laboratorio. In particolare, la tesi propone il passaggio da un modello legacy stateful, centrato su sequenze `SetValue`/`GetValue`, a un modello più esplicito e validabile, orientato a contratti dati e pipeline di esecuzione controllabili. La scelta metodologica è stata incrementale: mantenere la retrocompatibilità come vincolo non negoziabile e introdurre in parallelo un percorso moderno capace di migliorare robustezza, trasparenza e manutenibilità.

Dal punto di vista tecnico, il contributo principale riguarda la convergenza dei canali di trasporto su un flusso unico di normalizzazione, validazione ed esecuzione. Questo ha reso possibile introdurre funzionalità strutturali come JSON-run, validazione schema-based, gestione artifact persistente e supporto REST opzionale, evitando la proliferazione di percorsi applicativi divergenti. In parallelo, il driver Labber è stato riallineato al nuovo contratto server, con miglioramenti su parsing, gestione dello stato, naming/typing dei parametri e diagnostica degli errori.

La tesi evidenzia che il valore del refactoring non risiede solo nella qualità del codice, ma nell'affidabilità della pratica sperimentale che quel codice abilita. In un laboratorio quantistico, infatti, la qualità del dato dipende anche dalla qualità del software che orchestra le misure: ridurre ambiguità nei run, rendere espliciti input/output, tracciare artefatti e stabilizzare i fallback significa aumentare riproducibilità, auditabilità e confidenza nei risultati.

Il lavoro è stato accompagnato da una validazione progressiva su casi d'uso reali, checklist operative e test di regressione, con particolare attenzione a non rompere i workflow esistenti. I risultati mostrano una piattaforma più governabile: minor dipendenza da convenzioni tacite, migliore separazione delle responsabilità, estensioni più localizzate e rischio ridotto di effetti collaterali durante l'evoluzione. Il mantenimento del canale legacy come default ha inoltre favorito un'adozione graduale, compatibile con i vincoli quotidiani del laboratorio.

Resta un limite infrastrutturale, esplicitato nel lavoro: il completamento della validazione end-to-end sulla macchina clone è condizionato da policy IT e vincoli di rete esterni al software. Questo elemento non riduce la portata tecnica del refactoring, ma definisce con precisione il perimetro delle evidenze raccolte.

In prospettiva, la tesi individua tre direttrici di sviluppo: consolidamento delle validazioni sui nodi secondari; ulteriore standardizzazione della rappresentazione sperimentale per migliorare interoperabilità tra toolchain eterogenee; formalizzazione di benchmark continuativi (latenza, throughput, robustezza) tra modalità TCP e REST. Il risultato complessivo è il passaggio da uno strumento funzionale ma fragile a un'infrastruttura software documentabile, estendibile e sostenibile nel tempo, con impatto diretto sulla qualità del lavoro sperimentale e sulla trasferibilità delle pratiche nel contesto di ricerca.