

Progetto GAINS

(Generative AI for Network Sustainability)

Deliverable D2.2

Technical Report su Integrazione degli Agenti e Valutazione del Proof-of-Concept

Giordano, A. - Basile, L. - De Giacomo, A. - Haider, B. - Fruncillo, D. - McGann, O.

2026-01

Work Package: WP2 Progettazione del Sistema, Integrazione degli Agenti e Validazione del Proof-of-Concept

Stato: Definitivo | Versione: 1.0

Classificazione: Uso interno di progetto Condivisione controllata con partner su necessità

Abstract

Questo Technical Report documenta l'integrazione end-to-end degli agenti del sistema GAINS e la valutazione del proof-of-concept (PoC) come flusso completo di automazione e supporto decisionale in un contesto telco/cloud emulato, con estensioni su segnali reali in un segmento controllato di una rete telco reale direttamente collegata all'ambiente di ricerca costruito in datacenter. La finalità è dimostrare che l'interazione tra Network Monitor, Knowledge Manager e Network Orchestrator produce output operativamente utilizzabili, verificabili e coerenti con vincoli tecnici e standard, mantenendo al contempo efficienza operativa, scalabilità e presidi minimi di governance. Il perimetro della validazione include test funzionali e di stress su scenari modello, verifica di correttezza e consistenza degli artefatti prodotti configurazioni candidate, piani di change e rollback, report di validazione, pacchetti di explainability e valutazione dell'accuratezza e ripetibilità delle analisi ambientali, il tutto ancorato a un modello dati canonico e a contratti API versionati. Il PoC è costruito per essere replicabile in ambiente Docker e per poter essere validato da un team terzo senza accesso privilegiato a infrastrutture proprietarie: i componenti runtime sono containerizzati, le dipendenze sono esplicitate e configurate tramite variabili d'ambiente e secret management, e l'intero flusso è orchestrato da un Control Plane API-first che applica RBAC, tracciabilità end-to-end e audit trail append-only. La validazione è progettata per essere "black-box by contract": l'utente verifica comportamento e qualità osservando esclusivamente gli endpoint pubblicati, i payload canonici e gli artefatti persistiti, senza necessità di modificare il codice. Questo approccio riflette una premessa di trasferibilità industriale: i singoli agenti possono evolvere o essere sostituiti, purché contratti e modello canonico restino invariati e la catena trace_id, job, artifacts, audit, export manifest rimanga ricostruibile. L'integrazione si fonda su tre responsabilità operative distinte e intenzionalmente disaccoppiate. (i) Il Network Monitor opera in sola lettura e trasforma telemetria e storico in evidenze verificabili, baseline deterministiche, anomalie e indicatori strutturati; nel

PoC la sorgente di osservabilità è ancorata a un NMS già disponibile, così da accelerare l'end-to-end senza alterare invarianti architetturali e contratti. (ii) Il Knowledge Manager implementa un knowledge plane orientato all'operatività, in cui documenti, policy e artefatti vengono caricati nella base dati con pipeline deterministiche e resi interrogabili con retrieval ibrido e filtri vincolanti di applicabilità; le risposte operative sono grounded e corredate da citazioni verificabili, con meccanismi fail-closed o degradazione controllata quando le evidenze sono insufficienti. (iii) Il Network Orchestrator traduce un intent in configurazioni candidate contestualizzate e genera un job governato da stati espliciti, producendo artefatti versionati e hashati e introducendo un gating human-in-the-loop per approvazione o rifiuto in scenari a rischio operativo. La sostenibilità non è trattata come metrica "a posteriori", ma come oggetto canonico del workflow. Il PoC valida la produzione e l'uso di due oggetti chiave: FootprintInputs, generati dal Monitor come proxy di carico attribuibili per device e aggregabili per PoP, e FootprintEstimate, generata in orchestrazione come confronto as-is/to-be/delta con metodo dichiarato (metered, model_based, hybrid), base temporale coerente, assunzioni esplicite e livello di confidenza. La verifica end-to-end controlla che le stime degradino in modo deterministico e auditabile quando mancano segnali o dipendenze, e che ogni output esponga riferimenti alle evidenze utilizzate, rendendo la catena climate verificabile al pari della catena tecnica. La valutazione del PoC è strutturata in tre livelli complementari. Il primo livello è di conformità contrattuale: verifica che ogni endpoint critico rispetti schema, campi minimi obbligatori, error model standard e propagazione del trace_id, garantendo coerenza tra payload canonici e artefatti persistiti. Il secondo livello è funzionale e scenariale: su un insieme di scenari telco/cloud emulati, si misura se gli agenti cooperano correttamente nel produrre output coerenti, applicabili e verificabili, controllando in particolare (i) applicabilità del knowledge retrieval rispetto a vendor/ruolo/versione/servizio, (ii) presenza di citazioni robuste e assenza di claim non supportati in modalità operativa, (iii) correttezza strutturale e verificabilità dei piani di change e rollback, (iv) coerenza tra documentazione generata e artefatti tecnici consegnati. Il terzo livello è di robustezza e stress: si esercitano concorrenza, timeout, retry bounded, degradazione controllata in caso di dipendenza indisponibile e assenza di leakage in log e output; le prove includono job paralleli, controlli di idempotenza e validazioni su exportability, immutabilità e integrità degli artefatti. Il deliverable consolida i risultati sperimentali del task di integrazione e valutazione: (i) un ambiente PoC replicabile in Docker in cui i tre agenti interagiscono tramite contratti canonici e un endpoint esterno OpenAI-compatible configurabile, (ii) un harness di test e stress che abilita validazione automatizzata e raccolta evidenze con reporting strutturato, (iii) un export manifest che formalizza inventario e integrità del pacchetto evidenze tramite checksums e metadati d'ambiente, (iv) un set di metriche e quality gates per misurare qualità del retrieval e del grounding, qualità degli artefatti operativi e affidabilità della catena climate, (v) una sintesi di lessons learned focalizzate su determinismo, minimizzazione del rischio e riduzione delle ambiguità tra intent, policy e output tecnici. Infine, la validazione è progettata per essere fruibile anche da utenti non specialisti tramite una UI opzionale di demo e stress qualitativo che traduce interazioni conversazionali in chiamate verso il Control Plane, senza diventare una dipendenza strutturale del sistema e rimanendo confinata a supporto della sperimentazione.

Indice

1. Introduzione
 - 1.1 Obiettivi del Deliverable e del Proof-of-Concept
 - 1.2 Perimetro, assunzioni operative e vincoli
 - 1.3 Stakeholder, profili utente e responsabilità operative
 - 1.4 Definizioni operative e convenzioni intent, job, artifact, evidence, policy
2. Panoramica del PoC GAINS
 - 2.1 Componenti e responsabilità
 - 2.2 Flusso end-to-end di automazione e supporto decisionale
 - 2.3 Invarianti architetturali e principi di integrazione
 - 2.4 Oggetti canonici principali e relazioni logiche
3. Architettura di integrazione
 - 3.1 Vista logica a blocchi e confini di servizio
 - 3.2 Pattern di integrazione tra agenti
 - 3.3 Persistenza e versioning degli artefatti
 - 3.4 Osservabilità end-to-end logging, metriche, tracing
 - 3.5 Sicurezza di base e governance RBAC, audit, segregazione ambienti
4. Ambiente PoC e stack di esecuzione
 - 4.1 Prerequisiti software e hardware
 - 4.2 Struttura del workspace e repository layout
 - 4.3 Containerizzazione e networking
 - 4.4 Configurazione
 - 4.5 Avvio, arresto, reset e ripristino consistenza
 - 4.6 Diagnostica, troubleshooting e failure modes

5. Integrazione degli agenti
 - 5.1 Integrazione Network Monitor: ingressi, normalizzazione, output strutturati
 - 5.2 Integrazione Knowledge Manager: ingest, retrieval, citazioni, policy gating
 - 5.3 Integrazione Network Orchestrator: intent-to-plan, job state machine, artefatti
 - 5.4 Orchestrazione del flusso end-to-end e gestione dipendenze
 - 5.5 Idempotenza, retry bounded, timeout, rate limit e backpressure
 - 5.6 Human-in-the-loop e controlli di approvazione
6. Modello dati canonico e contratti API per la validazione
 - 6.1 Catalogo oggetti canonici e relazioni
 - 6.2 Schemi dei payload e campi obbligatori
 - 6.3 Regole di validazione e compatibilità di versione
 - 6.4 Error model standardizzato e semantica dei codici errore
 - 6.5 Tracciabilità end-to-end
 - 6.6 Export manifest e pacchetto di evidenze
7. Scenari modello telco/cloud per la valutazione
 - 7.1 Criteri di selezione e copertura funzionale
 - 7.2 Scenario S1: rilevazione anomalia, diagnosi, raccomandazione operativa
 - 7.3 Scenario S2: change su servizio, generazione candidate config, validazione e rollback plan
 - 7.4 Scenario S3: verifica compliance, vincoli e standard, generazione documentazione tecnica
 - 7.5 Scenario S4: ottimizzazione con vincoli di sostenibilità e confronto as-is/to-be
 - 7.6 Scenario S5: degradazione controllata e fail-closed in assenza di evidenze
 - 7.7 Dataset sintetici, seed e riproducibilità degli scenari

8. Piano di test end-to-end
 - 8.1 Obiettivi di test e strategia complessiva
 - 8.2 Matrice dei test
 - 8.3 Criteri di accettazione e quality gates
 - 8.4 Harness di test e automazione
 - 8.5 Raccolta evidenze, audit trail e criteri di completezza
 - 8.6 Gestione non conformità, issue taxonomy e triage
9. Test di stress, scalabilità e affidabilità
 - 9.1 Modello di carico e workload profile
 - 9.2 Concorrenza, throughput, latenza e saturazione risorse
 - 9.3 Soak test e stabilità nel tempo
 - 9.4 Fault injection e resilienza
 - 9.5 Idempotenza e consistenza degli artefatti sotto stress
 - 9.6 Risultati e interpretazione delle metriche di robustezza
10. Verifica della correttezza delle configurazioni e sicurezza operativa
 - 10.1 Validazioni sintattiche e semantiche
 - 10.2 Coerenza change plan, rollback plan e finestra operativa
 - 10.3 Guardrail e policy enforcement
 - 10.4 Minimizzazione rischio operativo e controlli di accesso
 - 10.5 Gestione secret e riduzione leakage in log e artefatti
11. Valutazione della qualità del Knowledge Plane
 - 11.1 Grounding e coverage delle citazioni
 - 11.2 Metriche retrieval e qualità evidenze
 - 11.3 Robustezza a prompt adversarial e allucinazioni operative
 - 11.4 Comportamento fail-closed e gestione incertezza

- 11.5 Coerenza tra documentazione generata e artefatti tecnici
- 12. Analisi ambientale e sostenibilità nel PoC
 - 12.1 Pipeline FootprintInputs: fonti, attributi, aggregazioni
 - 12.2 Stima FootprintEstimate: metodi, assunzioni e confidenza
 - 12.3 Coerenza temporale e confronti as-is/to-be/delta
 - 12.4 KPI operativi di sostenibilità
 - 12.5 Vincoli di sostenibilità e criteri di accettazione
- 13. Risultati della valutazione
 - 13.1 Risultati per scenario e copertura requisiti
 - 13.2 Risultati test end-to-end e qualità degli output
 - 13.3 Risultati stress e affidabilità
 - 13.4 Risultati su sostenibilità e consistenza delle stime
 - 13.5 Limiti osservati e condizioni di validità
- 14. Lessons learned
 - 14.1 Integrazione e contratti: cosa ha funzionato e cosa no
 - 14.2 Dati, determinismo e riproducibilità
 - 14.3 Operatività e sicurezza: gating e rischio residuo
 - 14.4 Sostenibilità: qualità input, stima e interpretazione
- 15. Conclusioni e sviluppi futuri
 - 15.1 Sintesi delle evidenze e livello di maturità del PoC
 - 15.2 Miglioramenti prioritari e roadmap di evoluzione
 - 15.3 Estensioni verso ambienti reali e scalabilità industriale

Bibliografia

Appendici

Appendice A. Guida rapida: esecuzione PoC in Docker

Appendice B. Tutorial: predisposizione di un endpoint esterno OpenAI-compatibile

Appendice C. Catalogo scenari e dataset sintetici

Appendice D. Schemi canonici e payload JSON completi

Appendice E. Checklist di validazione: comandi, attese, esiti attesi

Appendice F. Harness di test e stress: script, parametri, reporting e raccolta evidenze

Appendice G. Export manifest: struttura del pacchetto evidenze e verifica integrità

Appendice H. UI opzionale per demo e stress test: Gebo.ai

Appendice I. Matrice tracciabilità: requisiti - test - evidenze - artefatti

Appendice J. Glossario esteso e acronimi

Appendice K. Licenze, note di compliance e requisiti di redistribuzione