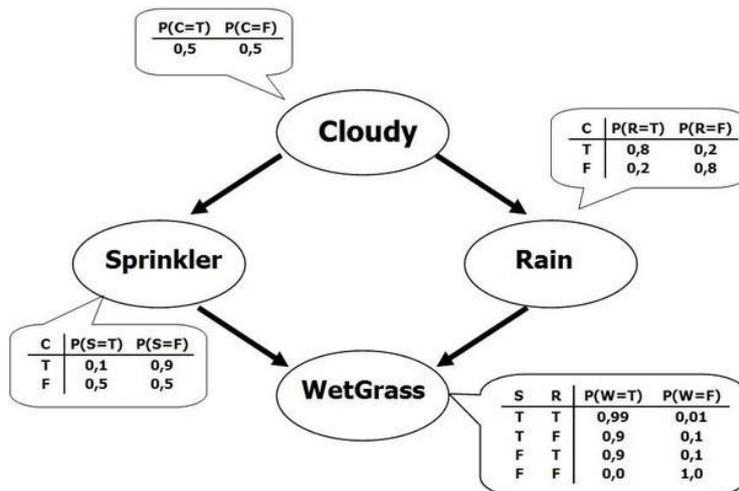


UNIVERSITÀ DEGLI STUDI DI PADOVA  
 Corso di Ingegneria del Software 2018/2019



# GRAFANA+BAYES

Monitoraggio per DevOps



# Oggetto dell'appalto

Oggi molte applicazioni vengono erogate come servizi nel Cloud. La Zucchetti, primaria software house italiana, da tempo produce procedure gestionali (ERP, Paghe, Fiscale) in tecnologia Web che vengono installate dai clienti nei Cloud pubblici, come Amazon AWS o Azure di Microsoft, nella server farm della Zucchetti stessa o in data center privati.

A partire dal prossimo anno tutte le fatture emesse in Italia dovranno transitare in formato elettronico dal Ministero dell'Economia e delle Finanze. La Zucchetti, principale operatore del settore, ha attivato un servizio di "Fatturazione Elettronica" nel proprio data center.

In questo caso il servizio è completamente gestito dalla Zucchetti che fa da tramite tra i propri clienti, che producono le fatture con i programmi gestionali, e il ministero, che le processa ed archivia.

Questo scenario ha preso il nome di "DevOps", dall'unione delle parole "Development" e "Operations": la fabbrica del software e gli operatori che erogano il servizio sono a stretto contatto tra loro e la costante collaborazione tra i due attori porta ad un notevole miglioramento della qualità, rispetto alle modalità tradizionali dove i due compartimenti sono fortemente separati.

Perché la collaborazione sia efficace è necessario un pieno monitoraggio dei sistemi, in modo che gli erogatori del servizio possano segnalare alla fabbrica i punti in cui è possibile migliorare la procedura. Nella modalità tradizionale i sistemisti si preoccupano solo dello stato del sistema, dovendo garantire la continuità del servizio, non del suo costante miglioramento.

La linea di produzione del software, così informata, può intervenire migliorando i punti critici che l'erogazione mette in evidenza. Ne consegue che i prodotti seguiti in "DevOps" hanno molti più rilasci, vicini nel tempo, e risultano di qualità superiore.

Per eseguire il monitoraggio dei propri sistemi la Zucchetti ha scelto Grafana, un prodotto Open Source, individuato come il miglior sistema di monitoraggio. Grafana è estendibile con plug-in in linguaggio JavaScript.

Gli elementi oggi presenti in Grafana sono sistemi di presentazione del dato, raccolti in Dashboard operative, e sistemi di allarme al raggiungimento di determinate soglie.

Per dispiegare i vantaggi del "DevOps" la Zucchetti vorrebbe realizzare dei sistemi che possano applicare metodi di intelligenza artificiale al flusso dei dati raccolti, al fine non solo di monitorare la "liveliness" del sistema ma anche per consigliare gli interventi o quanto meno le zone di intervento alla linea di produzione del software.

La tecnica che appare più promettente è quella delle reti Bayesiane, perché la competenza degli esperti viene raccolta in un sistema formale probabilistico che poi può essere collegato ai dati effettivamente raccolti sul campo per determinare quali eventi non visibili siano i più probabili.



Il presente capitolato ha per oggetto l'affidamento della fornitura per la realizzazione di un plug-in per lo strumento di monitoraggio Grafana che applichi reti Bayesiane al flusso dei dati ricevuti per allarmi o segnalazioni tra gli operatori del servizio Cloud e la linea di produzione del software.

## Caratteristiche e Requisiti Obbligatori

Il sistema che richiediamo sarà un plug-in di Grafana, scritto in linguaggio JavaScript, che leggerà da un file json la definizione della rete Bayesiana e quindi permetterà di associare ad alcuni nodi della rete dati prelevati dal flusso del monitoraggio.

Ad intervalli predefiniti o con continuità, verranno eseguiti i calcoli previsti dalla rete Bayesiana, modificando le probabilità dei nodi derivati in base ai dati rilevati dal campo.

Il software richiesto dovrà svolgere almeno i seguenti compiti:

1. Leggere la definizione della rete Bayesiana da un file in formato json.
2. Associare dei nodi della rete, letta dal file json, ad un flusso di dati presente in Grafana.
3. Applicare il ricalcolo delle probabilità della rete secondo regole temporali prestabilite.
4. Fornire nuovi dati al sistema di Grafana derivati dai nodi della rete non collegati al flusso di monitoraggio.
5. Rendere disponibili i dati al sistema di creazione di grafici e dashboard per la loro visualizzazione.



## Requisiti Opzionali

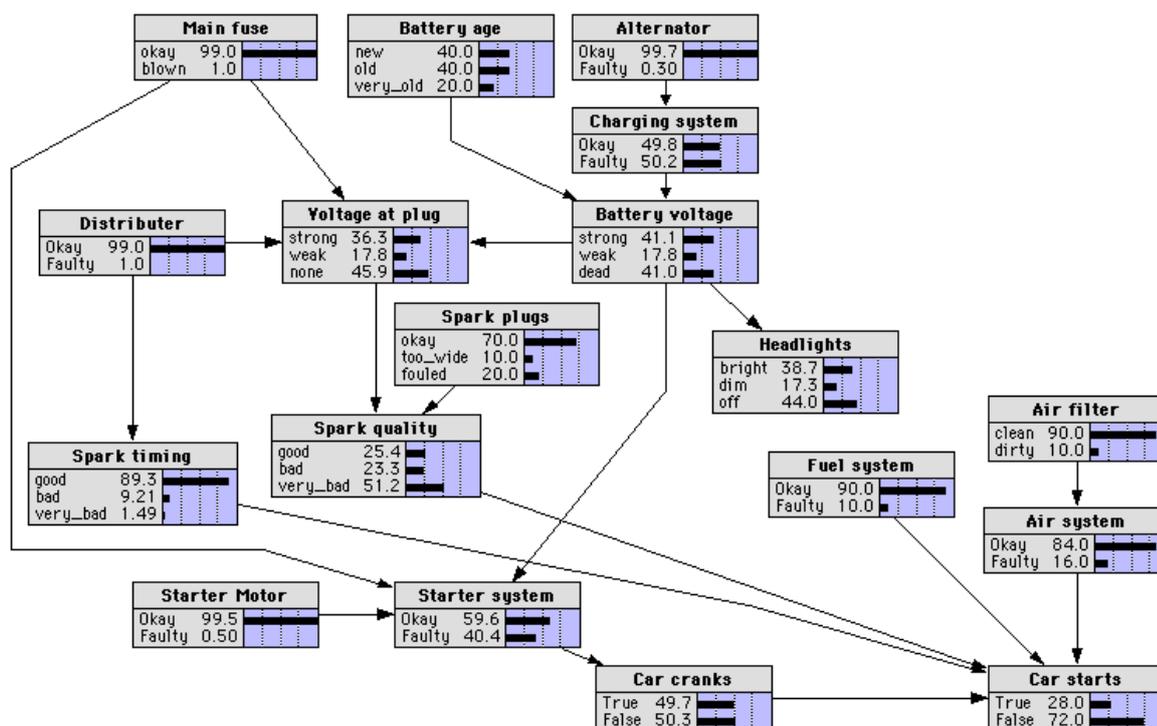
Il software potrebbe avere le seguenti caratteristiche:



1. Possibilità di definire “alert” in base a livelli di soglia raggiunti dai nodi non collegati al flusso dei dati.
2. Possibilità di disegnare la rete Bayesiana con un piccolo editor grafico specializzato.
3. Possibilità di applicare più reti Bayesiane in oggetti di monitoraggio diversi.
4. Possibilità di creare una rete Bayesiana a partire dai dati raccolti sul campo anziché svilupparla con la collaborazione degli esperti del settore.
5. Identificare altri metodi di Intelligenza Artificiale oltre alla rete Bayesiana che siano applicabili all’analisi del flusso di dati di monitoraggio al fine di aiutare la linea di produzione del software a scoprire i punti migliorabili dalla propria procedura.

## Suggerimenti

L’immagine seguente riporta una rete Bayesiana applicata al problema dell’avvio di una automobile. Al livello più alto si vedono tre fatti: lo stato del fusibile, l’età della batteria e lo stato dell’alternatore. La probabilità che l’alternatore sia rotto “a priori” è del 0.30%, mentre quella che sia funzionante è del 99.7%. La batteria, sempre a priori secondo l’esperto, ha il 40% di probabilità di essere nuova, il 40% di essere vecchia, ed il 20% molto vecchia. Il fusibile al 99% è OK, all’ 1% è saltato. Questi stati iniziali influenzano i possibili stati composti, per cui un alternatore rotto può portare ad una batteria scarica attraverso il nodo “charging system”, per cui nel nodo che indica il “Battery voltage” troviamo il 41.1% di probabilità che sia forte (strong), il 17.8% che sia debole (weak) e il 41% che la batteria non dia segni di vita (dead).



Proseguendo nell'immagine si arriva fino al nodo finale marcato con "Car starts" che indica che la macchina riesce a mettersi in moto con una probabilità del 28%, mentre vi è una probabilità del 72% che non riesca a partire.

La rete Bayesiana si usa indicando quale fatto di interesse possiamo osservare nella realtà; per esempio, potremmo indicare che stiamo osservando un "Battery voltage" di tipo "weak". A questo punto le probabilità di quel nodo cambiano in 0% per "strong", 100% per "weak" e 0% per "dead". Si possono quindi ricalcolare tutte le probabilità di conseguenza, sia verso l'alto che verso il basso. In questa condizione per esempio diminuirà la probabilità che la macchina si metta in moto e aumenteranno le probabilità che la batteria sia vecchia e l'alternatore rotto.

Il programma che viene proposto dovrà comportarsi come in questo esempio: una rete Bayesiana di verifica dell'applicazione verrà caricata su Grafana, fornendo informazioni sullo stato del sistema. Le probabilità verranno quindi ricalcolate in base ai fatti effettivamente osservati, facendo salire o scendere le probabilità di fatti come query poco performanti, procedure lente, utenti con attività inaspettata e così via.

Ad esempio, un gran numero di accessi al disco normalmente indica query non ottimizzate; di solito, aggiungendo indici opportuni questa "malattia" viene curata. La rete Bayesiana potrebbe accendere un "alert" di probabili indici mancanti nella dashboard di osservazione del sistema quando rileva la situazione descritta, mentre il monitoraggio semplice indica solo la presenza di attività su disco senza indagare le possibili cause.

Per la gestione dei calcoli della rete Bayesiana indichiamo una libreria Open Source che è già stata da noi utilizzata con successo in altre applicazioni.

La libreria può essere liberamente scaricata all'indirizzo: <https://github.com/vangi/jsbayes>

Per comprendere l'uso delle reti Bayesiane può essere di aiuto il software di disegno e calcolo scaricabile all'indirizzo: <http://unbbayes.sourceforge.net/>

Il prodotto di monitoraggio Grafana è reperibile all'indirizzo: <https://grafana.com/>

## Variazioni ai requisiti

In corso d'opera non sarà possibile variare/modificare i requisiti minimi (obbligatorie per accettare il prodotto). Sarà invece possibile variare i requisiti opzionali, in quanto saranno i gruppi vincitori dell'appalto a modificarli / eliminarli / aggiungerli.

## Documentazione

Il progetto dovrà essere supportato dalla documentazione minima richiesta per il corso di Ingegneria del software e dovrà essere fornito un manuale per l'utilizzo ed un manuale per chiunque voglia estendere l'applicazione.



## Garanzia e Manutenzione

L'azienda Zucchetti SPA è interessata a questo progetto come dimostrazione della fattibilità dell'obiettivo utilizzando le tecnologie web. Costituirà titolo preferenziale nella valutazione delle proposte la pubblicazione del progetto sul sito “github.com” o altri repository pubblici, in conformità con i relativi requisiti di natura open-source, per favorire la continuità del prodotto risultante.

## Rinvio

Per tutto quanto non previsto nel presente capitolato, sono applicabili le disposizioni contenute nelle leggi e nei collegati per la gestione degli appalti pubblici.