Attentato a Ranucci, Carotenuto: “Governo mandante morale, l’ho detto e lo ridirei”

(Adnkronos) –
“Io quella cosa la ribadirei anche oggi”. Lo afferma all’Adnkronos il deputato del Movimento 5 Stelle ex componente della Commissione parlamentare di Vigilanza Rai, Dario Carotenuto, tornando sul post pubblicato dopo l’attentato a Sigfrido Ranucci, nel quale definiva il governo “mandante morale”.

“Anche se fosse vero, come sta emergendo, che Valter Lavitola abbia pensato di fare quasi un favore a Ranucci, all’insaputa di Ranucci, io continuo a dire che, se a Lavitola viene in mente una cosa del genere, è proprio perché la destra ha attaccato tutti i giorni Ranucci in quel modo. Io non posso che ribadire quello che ho scritto in quel post”, afferma il parlamentare pentastellato. “Per come sono oggi le cose, mi sembra quasi che stiano confermando quella mia ipotesi, altro che sconfessarla”.

Carotenuto precisa però il significato delle sue parole: “È chiaro che non ho detto che hanno messo la bomba o che qualcuno l’abbia messa su indicazione di Meloni. Non mi è passato mai neanche lontanamente per la testa”. Secondo il deputato M5S, il riferimento era invece “al clima di tensione e di delegittimazione” nei confronti del conduttore di Report: “Quello che stavo osservando da quando avevo messo piede in Vigilanza era un attacco quotidiano da parte della destra a Ranucci e alla sua trasmissione. Sto dicendo semplicemente che si è creato un clima tale che poi a qualche mente squilibrata passa per la testa di fare una cosa del genere. Questo

intendevo quando parlavo di mandante morale”.

Il Giornale fa riferimento ad una conversazione tra il deputato e Ranucci. Nella conversazione, verrebbe fatto il nome di Gimmi Cangiano, deputato di Fratelli d’Italia: “L’articolo parla di ‘intercettazione shock’, ma è un titolo che poi, secondo me, non rispecchia quello che si legge nel testo. Dov’è lo shock? Io, anzi, mettevo esplicitamente in dubbio che Gimmi Cangiano fosse implicato” nell’attentato contro Ranucci.

Carotenuto ricostruisce il contesto della telefonata: “C’era stata una puntata di Report nella quale ci si interrogava, nel quadro più ampio dell’inchiesta, su un eventuale ruolo di Gimmi Cangiano. Io, conoscendolo, seppur non benissimo, quella ricostruzione la ritenevo poco verosimile. Quando ho sentito Sigfrido gli ho detto: ‘Ma tu veramente pensi che ci possa essere Cangiano?’. Mi sembrava un’ipotesi fuori dal verosimile”. Il deputato precisa di aver voluto offrire al giornalista la propria valutazione personale: “È stato un mio impulso dire che, secondo me, Cangiano non c’entrava. Anche per dare una mano a Ranucci, che immaginavo stesse cercando di capire chi gli avesse fatto l’attentato, volevo dargli dalla mia prospettiva un input: per quelle che erano le mie conoscenze, secondo me Cangiano non c’entra”.

Quanto al rapporto con il conduttore di Report, Carotenuto esclude retroscena: “Dopo quella telefonata non ci siamo mai più visti per parlarne. Ci siamo incrociati soltanto in un’occasione pubblica. Ogni tanto ci scriviamo su WhatsApp, ma molto raramente”. E ribadisce la propria stima: “Io considero Sigfrido una vittima in questa vicenda, assolutamente una vittima. Gli esprimo vicinanza e lo stimo come giornalista. Da ragazzo sognavo perfino di lavorare a Report”.

Il deputato M5S torna anche sulle polemiche seguite a un suo post in cui definiva il governo “mandante morale” dell’attentato: “Non ho mai detto che qualcuno a Palazzo Chigi

abbia ordinato la bomba, non mi è passato neanche lontanamente per la testa. Io parlavo del clima che si era creato attorno a Ranucci e a Report, di un attacco quotidiano e di una delegittimazione continua. È questo che intendevo quando parlavo di ‘mandante morale’: un clima che può aver favorito il gesto di qualche mente squilibrata, non certo un mandato politico”.

–

politica

webinfo@adnkronos.com (Web Info)

Novità su Google, per aggiungere Key4Biz tra le tue fonti preferite, clicca qui



Leggi le altre notizie sull’home page di Key4biz

https://www.key4biz.it/attentato-a-ranucci-carotenuto-governo-mandante-morale-lho-detto-e-lo-ridirei/581298/

Canada, oltre 200 incendi fuori controllo: il fumo ‘invade’ anche gli Usa, ira

di Trump

(Adnkronos) – Oltre 200 incendi boschivi sono fuori controllo in Canada, in particolare nella provincia dell'Ontario, da dove parte il fumo che 'invade' il nord-est degli Stati Uniti. Secondo gli ultimi dati del Centro interagenzie canadese per gli incendi forestali (Ciffc), 207 incendi sono fuori controllo in tutto il paese, su un totale di 897 incendi attivi. Quasi 2,8 milioni di ettari sono andati in fumo dall'inizio dell'anno, secondo gli ultimi dati ufficiali del governo federale. La situazione è particolarmente critica in Ontario, dove oltre 80 incendi boschivi sono fuori controllo, ha dichiarato il premier Doug Ford ai giornalisti.

Ieri l'Ontario ha richiesto assistenza federale e sta ricevendo supporto dai vigili del fuoco di altre province, tra cui Alberta e Yukon. Più di 80 aerei antincendio ed elicotteri sono stati dispiegati in Ontario per combattere le fiamme. Altri 39 velivoli sono pronti a decollare verso le zone remote inaccessibili via terra per evacuare le comunità abitate principalmente da popolazioni indigene. Finora, gli incendi non hanno causato vittime.

Il fumo degli incendi, trasportato dal vento, ha causato un inquinamento atmosferico estremo nella Greater Toronto Area, la città più popolosa del Canada, e negli Stati Uniti orientali. Secondo i dati raccolti dalla società svizzera IQAir, Detroit, Chicago e Washington erano tra le città più inquinate al mondo oggi. Il Servizio Meteorologico Canadese ha avvertito che la qualità dell'aria a Toronto, che era "migliorata" stamattina, "tornerà a essere molto scarsa" in serata e potrebbe rimanere tale per tutto il fine settimana.

Il presidente americano Donald Trump ha detto intanto di "ritenere il Canada responsabile" per gli incendi che si stanno verificando nel Paese, affermando che "è assolutamente inaccettabile" che il fumo causato dalle fiamme si propaghi

negli Stati Uniti. ”Riteniamo il Canada responsabile del fatto che non stia mantenendo adeguatamente le proprie foreste e la vegetazione in esse contenuta, e che gli Stati Uniti vengano inutilmente invasi da aria sporca, inquinata e malsana, la cui qualità è pericolosa e assolutamente inaccettabile!”, ha scritto Trump su Truth Social.

”Chiamerò il primo ministro durante la giornata per sapere cosa intendono fare al riguardo. Il costo è incalcolabile!”, ha proseguito Trump sostenendo che ”il Canada si è rifiutato di impegnarsi in una gestione forestale di base e nella rimozione dei detriti, sapendo che tale rifiuto porterà esattamente a questo risultato. Questa è negligenza intenzionale, che sta diventando un evento annuale, costando agli Stati Uniti miliardi di dollari, costo che questo inquinamento deve necessariamente essere aggiunto alle tariffe che il Canada sta attualmente pagando”, ha aggiunto.

–

internazionale/esteri

webinfo@adnkronos.com (Web Info)

Novità su Google, per aggiungere Key4Biz tra le tue fonti preferite, clicca qui



Leggi le altre notizie sull’home page di Key4biz

https://www.key4biz.it/canada-oltre-200-incendi-fuori-controllo-il-fumo-invade-anche-gli-usa-ira-di-trump/581284/

Mps-Banco Bpm, come si arriva a Generali (e Axa)? Cosa c’è dietro il progetto di fusione

(Adnkronos) – Cosa ci sarebbe dietro il progetto di una fusione tra Mps e Banco Bpm, che ovviamente coinvolge Credit Agricole Italia? E perché si può arrivare a Generali e Axa? L’indiscrezione raccolta dall’Adnkronos può essere letta da diversi punti di vista. Un buon punto di partenza può essere la lettura del comunicato diffuso da Mps a valle del consiglio del 16 Luglio, in cui di fatto si esprime una preferenza verso l’opzione di fusione con BPM. La conseguenza di questa operazione sarebbe avere Credit Agricole come primo socio della banca che nascerebbe dalla fusione, con una quota che le diverse ricostruzioni collocano tra l’11% e il 12% del capitale complessivo, comunque la partecipazione singola più rilevante.

Dal punto di vista politico, questo scenario pone una questione non secondaria. Un anno fa il governo è intervenuto esercitando il golden power per condizionare l’offerta di Unicredit su BPM, con l’obiettivo dichiarato di proteggere l’italianità del sistema bancario. Il paradosso, sollevato da alcuni osservatori, è che quell’intervento potrebbe aver finito per consolidare proprio il ruolo di Crédit Agricole, oggi arbitro di fatto del futuro di Piazza Meda.

Gli effetti dell’integrazione tra Mps e Banco Bpm non si fermano però qui. Alle spalle di Crédit Agricole, il secondo azionista della futura aggregazione sarebbe Delfin, la holding lussemburghese della famiglia Del Vecchio, presieduta da Francesco Milleri. Delfin è oggi il primo socio privato di Mps (quota vicina al 17,5%), secondo azionista di Generali (circa il 10%) e detiene inoltre il 2,7% di Unicredit: una presenza trasversale che la rende, di fatto, un convitato di pietra in

ogni tavolo del risiko bancario.

Andando ancora oltre, molti fili della tela di Delfin conducono a Parigi. Il principale socio di Delfin in Essilor Luxottica è lo Stato Francese, che è entrato nel capitale del colosso degli occhiali per sostenere l'azienda a lungo termine. Questo è avvenuto tramite due principali enti pubblici: Bpifrance (la banca pubblica d'investimento) e la Caisse des Dépôts, la CdP francese. Non solo, nel pool di banche finanziatrici di Leonardo Maria Del Vecchio, figlio del fondatore di Luxottica, ci sono sia Crédit Agricole sia BNP Paribas: due tra i maggiori gruppi bancari francesi.

C'è un altro tassello da aggiungere al mosaico. Non va dimenticato che l'intera partita ha come posta in gioco anche la quota del 13% circa in Assicurazioni Generali che Mps ha ereditato con l'acquisizione di Mediobanca, una quota capace di orientare gli equilibri di governance del Leone di Trieste. Nel caso in cui si dovesse arrivare ad avere Credit Agricole primo socio della banca che controlla anche quel 13% in Generali, potrebbero trovare una sponda affidabile anche le mire di Axa sul Leone di Trieste. A quel punto la presenza francese nel sistema bancario e assicurativo italiano avrebbe un peso ancora più rilevante.

—

economia

webinfo@adnkronos.com (Web Info)

Novità su Google, per aggiungere Key4Biz tra le tue fonti preferite, clicca qui



Leggi le altre notizie sull'home page di Key4biz

https://www.key4biz.it/mps-banco-bpm-come-si-arriva-a-generali-e-axa-cosa-ce-dietro-il-progetto-di-fusione/581286/

Apple Music is getting a price hike



Apple Music is more expensive now. In the US, an individual plan now costs $11.99 per month, a $1 bump up from the previous $10.99 price. A family plan now costs $19.99 per month, up from $16.99, and a student plan costs $6.99 per month, up from $5.99. Apple, in a statement to *Music Business Worldwide*, says it is increasing prices “as a result of rising licensing costs.”

Music Business Worldwide also reports that prices gone up in

the UK and Europe, and the publication “understands” that hikes are hitting other countries as well. Apple didn’t immediately reply to a request for comment from *The Verge*.

Apple Music last got a price increase in October 2022 alongside hikes to Apple TV Plus and Apple One. The company is also reportedly raising prices for AppleCare Plus subscriptions for Macs and iPads. Spotify recently increased its prices in the US as well, hiking up a Spotify Premium plan from $11.99 per month to $12.99 per month. In other music industry news, CD sales in the US have reportedly been going up.

Apple has also increased some prices of Apple One subscriptions, *MacMagazine* reports. In the US, the Family plan has increased to $27.95 per month and the Premier plan has increased to $39.95, both hikes of $2 from the previous costs. The individual plan remains $19.95 per month.

***Update, July 17th**: Added Apple One price hikes.*

https://www.theverge.com/tech/967379/apple-music-price-increase

---

Cina, frana a Chongqing: almeno 8 morti e 34 dispersi

(Adnkronos) – Una frana ha travolto la contea di Pengshui, nella municipalità cinese di Chongqing, provocando almeno otto vittime e lasciando 34 persone disperse. Secondo quanto riferito dall’emittente statale Cctv, il cedimento si è verificato intorno alle 9:10 ora locale (le 3:10 in Italia), interessando una zona residenziale e commerciale ai piedi di

una montagna. Oltre 800 soccorritori sono stati mobilitati per le operazioni di emergenza: 18 persone rimaste intrappolate sono state estratte dalle macerie, ma otto di loro sono state dichiarate morte.

Il presidente cinese Xi Jinping ha ordinato alle autorità di accelerare gli accertamenti sulle cause del disastro e di “individuare ed eliminare i rischi di calamità geologiche e altri potenziali pericoli”. L’area colpita è caratterizzata da un terreno montuoso e ripido, definito da un funzionario locale come “imprevedibile”, con la presenza di rocce ancora instabili lungo i versanti. Il governo ha stanziato 50 milioni di yuan (oltre 7 milioni di euro) per sostenere le operazioni di soccorso e fornire assistenza ai residenti colpiti. La tragedia arriva meno di due settimane dopo un’altra frana nella provincia nord-occidentale del Gansu, dove 21 delle 33 persone coinvolte avevano perso la vita.

–

internazionale/esteri

webinfo@adnkronos.com (Web Info)

Novità su Google, per aggiungere Key4Biz tra le tue fonti preferite, clicca qui



Leggi le altre notizie sull’home page di Key4biz

https://www.key4biz.it/cina-frana-a-chongqing-almeno-8-morti-e-34-dispersi/581276/

Mps, allo studio matrimonio con Banco Bpm: il progetto potrebbe favorire un’operazione Generali-Axa

(Adnkronos) – Un piano di fusione a cascata. Potrebbe essere questo – secondo quanto apprende Adnkronos da fonti finanziarie qualificate – il progetto a cui starebbero lavorando Luigi Lovaglio, amministratore delegato di Mps, e Vittorio Grilli, presidente di Mediobanca, in risposta all’opas lanciata da Intesa Sanpaolo. La fusione – sempre secondo quanto si apprende – coinvolgerebbe Mps, Banco Bpm e indirettamente Credit Agricole Italia. L’operazione allo studio avrebbe come ulteriore sviluppo l’apertura di un dossier di fusione tra Generali e Axa. Da alcune settimane sarebbe in corso il dialogo tra i vertici di Banco Bpm e quelli della Banque Verte. Il progetto, viste le posizioni già espresse pubblicamente sulla nascita di un terzo polo bancario, potrebbe trovare il sostegno dell’area leghista nella maggioranza, che ha sempre visto con favore la fusione tra Piazza Meda e Siena. (Di Andrea Persili)

–

economia

webinfo@adnkronos.com (Web Info)

Novità su Google, per aggiungere Key4Biz tra le tue fonti preferite, clicca qui



Leggi le altre notizie sull’home page di Key4biz

https://www.key4biz.it/mps-allo-studio-matrimonio-con-banco-bpm-il-progetto-potrebbe-favorire-unoperazione-generali-axa/581288/

Apple’s plot to crush OpenAI



Apple is suing OpenAI. The complaint is readable and intense, as these things often are, though many experts seem to think many of the allegations are just the ways things are done. So what does Apple really want here, and why is it picking such a public fight with OpenAI?

On this episode of *The Vergecast*, Nilay and David go through the lawsuit, and look at Apple’s history of splashy litigation to determine whether Apple is worried about a possible

competitor or simply looking to capitalize on a weak moment for OpenAI. All this is happening as Apple ships the public betas of its new software, headlined by the new Siri AI, and we have thoughts about what it all means – and whether the new Siri is actually any good.

https://www.theverge.com/podcast/967244/apple-openai-lawsuit-vergecast

---

Traghetti, aumentano i prezzi in Italia

(Adnkronos) – Nel secondo trimestre del 2026 i prezzi dei traghetti in Italia sono aumentati del 7,4% su base annua, con una tariffa media di 30,70 euro: i dati

–

multimedia/news-to-go

webinfo@adnkronos.com (Web Info)

Novità su Google, per aggiungere Key4Biz tra le tue fonti preferite, clicca qui



Leggi le altre notizie sull'home page di Key4biz

https://www.key4biz.it/traghetti-aumentano-i-prezzi-in-italia/581290/

Ticker: Today Show Intruder Charged With Hate Crime After Shouting Slurs at Craig Melvin



Top of the Ticker: Thursday's edition of NBC News' morning show, Today, appeared to run smoothly, but behind the scenes, an unsettling situation took place as an intruder made his way into the secure production area.

During a break in the show, **Andrew Truelove** allegedly managed to bypass security at NBC News' offices at 30 Rockefeller and find his way to the street-level studio where he confronted Today co-anchor **Craig Melvin** and hurled racial slurs. According to a statement provided to TVNewser by NBC News, Melvin alerted security, and Truelove was detained and taken into custody by law enforcement without further incident.

On Friday, Truelove, according to a complaint filed by the Manhattan District Attorney's Office, was charged with hate

crime burglary, hate crime menacing, hate crime criminal trespass, and harassment. The judge assigned to the case set bail at $10,000.

In the NBC News statement, the network said it takes the safety and security of its employees and guests “extremely seriously” and is reviewing the incident and its security protocols, while remaining committed to providing a safe and secure environment for all who work or visit the studios.

Meanwhile, Melvin, on Friday’s broadcast, addressed the incident, saying they are fully cooperating with the NYPD during the investigation and adding that “we are just happy that everyone is safe.”

New Suit: Two Utah-based nonprofits have filed a lawsuit against Fox News Media and reality star and businessman **Kevin O’Leary**, alleging defamation after O’Leary, who appeared on the network in May, claimed these organizations were “proxies for the Chinese government.” The nonprofit founders claimed they have suffered reputational, financial, and emotional damage after O’Leary repeated these allegations at least 10 times on Fox News, according to the suit.

In a statement to TVNewser, Fox News Media said it “publicly corrected the record on every program where on-air guest Kevin O’Leary’s comments were made, all of which was extensively publicized. We will vigorously defend against this lawsuit.”

Champagne toast: MS NOW celebrated its 30th anniversary this week, marking the occasion with a soiree featuring current and former executives. As part of the anniversary celebrations, the network will be heading to the Dallas-Fort Worth metro area for its annual fan event, MS NOW Live: Together in Texas. The community event will take place at College Park Center at the University of Texas at Arlington on Saturday, September 26, putting the network, its journalists, newsmakers, and special guests in the home of one of the most hotly contested

senate races ahead of the midterm election.

Election prep: Speaking of the midterm elections, CNN revealed it will be hosting the Maine Democratic Senate Debate on Thursday, July 23, at 8 p.m. ET. In partnership with the Bangor Daily News, the two-hour debate will be moderated by CNN anchors **Jake Tapper** and **Dana Bash**, along with **Michael Shepherd**, the Bangor Daily News' political director.

Revolving Door: Former The Washington Post scribe **Brittany Shammas** is joining CNN as a reporter for the national news team. Based in Washington, D.C., Shammas will focus on distinctive national reporting and storytelling that goes beyond the headlines.

Meanwhile, MS NOW announced that **Brian Swarth** is joining the network as a senior vice president of marketing. Swarth comes to the network from AMC Networks, where he was a svp of product strategy and business operations. In this role, Swarth will oversee audience growth strategies, consumer engagement, and brand marketing across MS NOW's portfolio, including linear television, digital platforms, direct-to-consumer products, live events, and more.

Recommended videos
https://www.adweek.com/tvnewser/today-show-intruder-charged-with-hate-crime-after-shouting-slurs-at-craig-melvin/

MQTT – come usare MQTT in C – pt.3

Logan: *Lo sai che sono tutte balle, vero? Sarà successo un*

quarto di questa roba e non così. E nel mondo reale le persone muoiono e nessun presuntuoso in calzamaglia del cazzo può evitarlo!

E siamo giunti, finalmente, alla terza e ultima puntata di *"MQTT – come usare MQTT in C"*. E visto che la coerenza per me è molto importante, in questo articolo mostrerò come usare **MQTT** in **Go** (*ah ah ah*).

Beh, come ben sanno i miei lettori più assidui, io sono, fondamentalmente, un fanatico del C, però uso spesso (per lavoro) anche il Go e non disdegno di parlarne nei miei articoli; sarà, magari, perché è un bel linguaggio *C-like*, creato da due storici (e mitici) *guru* del mondo UNIX, **Ken Thompson** e **Rob Pike**, affiancati dal bravissimo **Robert Griesemer** (anche lui proveniente dal mondo UNIX). E non solo: il Go è la risposta di questi tre grandi informatici (e di Google) alle esagerate complicazioni introdotte nel C++11 (*e non avevano ancora visto i successivi C++14/17/20/23!!!*). In fondo all'articolo, per chi ha voglia di leggerlo, citerò uno scritto di **Rob Pike** sulla genesi del Go (che ho già citato **qui** e **qui).**

E la frase di **Logan** qui sopra? Beh, io direi che calza a pennello: *nel mondo reale (della programmazione) bisogna fare scelte oculate (fregandosene delle mode) e il Go, per alcuni progetti, è una scelta oculata.*



…meno balle: oggi usiamo il Go!…

Orbene, veniamo al dunque: ho scritto un programma *mqttclient.go* che è una replica delle due versioni in C pubblicate **qui** e **qui.** L'idea era che si potesse eseguire esattamente nello stesso modo e, ovviamente, produrre gli stessi risultati. Ok, vediamo un po se ci sono riuscito. Vai col codice!

```
// mqttclient.go - un semplice client mqtt in Go
package main import ( "fmt" "time" "os" mqtt "github.com/eclipse/paho.mqtt.golang"
) // funzione main
func main() { // test del numero di argomenti argc := len(os.Args) if (argc != 4) { // errore: numero errato di argomenti fmt.Printf("%s: numero errato di argomenti\n", os.Args[0]) fmt.Printf("uso: %s clientId topicPub topicSub (e.g.: %s client1 test/topic2 test/topic1)\n", os.Args[0], os.Args[0]) os.Exit(1) } // set delle opzioni di connessione opts := mqtt.NewClientOptions() opts.AddBroker(fmt.Sprintf("tcp://%s:%d", "localhost", 1883)) opts.SetClientID(os.Args[1]) opts.SetDefaultPublishHandler(onMessage) opts.OnConnect = onConnect opts.OnConnectionLost = onConnectionLost // mi collego client := mqtt.NewClient(opts) if token := client.Connect(); token.Wait() && token.Error() != nil {
```

```
panic(token.Error()) } // eseguo subscribe e publish subscribe(client) publish(client) // mi disconnetto time.Sleep(2 * time.Second) client.Disconnect(250)
} // publish - funzione per pubblicare sui topic
func publish(client mqtt.Client) { // pubblico messaggi sull'altro topic (topicPub in "uso: %s clientId topicPub topicSub") for i := 0; i <= 2; i++ { // attendo 2 secondi prima di pubblicare un messaggio time.Sleep(2 * time.Second) // pubblico text := fmt.Sprintf("messaggio %d dal client %s", i, os.Args[1]) token := client.Publish(os.Args[2], 0, false, text) token.Wait() fmt.Printf("Messaggio %d publicato sul topic %s\n", i, os.Args[2]); }
} // subscribe - funzione per sottoscriversi ai topic
func subscribe(client mqtt.Client) { // mi sottoscrivo a un topic (topicSub in "uso: %s clientId topicPub topicSub") token := client.Subscribe(os.Args[3], 1, nil) token.Wait() fmt.Printf("%s: mi sottoscrivo al topic %s\n", os.Args[1], os.Args[3]);
} // onMessage - una callback per i messaggi che arrivano sul topic sorvegliato
var onMessage mqtt.MessageHandler = func(client mqtt.Client, msg mqtt.Message) { fmt.Printf("messaggio ricevuto sul topic %s: \"%s\"\n", msg.Topic(), msg.Payload())
} // onConnect - una callback per gestire la connessione
var onConnect mqtt.OnConnectHandler = func(client mqtt.Client) { //fmt.Println("Connected")
} // onConnectionLost - una callback per gestire la disconnessione
var onConnectionLost mqtt.ConnectionLostHandler = func(client mqtt.Client, err error) { fmt.Printf("Connection lost: %v", err)
}
```

Come avrete notato, il codice è abbastanza semplice e **super-commentato**, quindi con c'è molto da spiegare, anche se per i lettori completamente a digiuno del Go non tutto sarà chiarissimo… ma questo non è un corso rapido di Go (*ah ah ah*). Vi do un consiglio spassionato: se vi interessa ampliare i vostri orizzonti e avere, anche, nuovi e interessanti sbocchi nel mondo del lavoro, studiatevi il Go! Si può cominciare con l'ottimo corso **Go by Example**, che è gratis ed è proposto e

curato direttamente dal **team ufficiale del Go** (io ho cominciato, molti anni fa, proprio con questo corso e poi ho proseguito come autodidatta).

Chiusa la parentesi posso dire che, a parte eventuali incomprensioni da linguaggio, se fate un confronto con gli analoghi Client scritti in C dei precedenti articoli, noterete che il codice è veramente molto simile ed è moderatamente più compatto, visto che il Go è, praticamente, un C ad alto livello.

Faccio nuovamente notare che anche questo esempio si collega a un *broker* locale (ossia installato sulla macchina stessa di esecuzione), quindi il collegamento è fatto con l'indirizzo: *"tcp://localhost:1883"*. Ma ci sono anche dei broker pubblici disponibili in rete che funzionano esattamente come quello locale. Nel mio caso (Linux) il *broker* locale che uso è **Mosquitto** (disponibile anche per Windows) e funziona alla grande.

Ed è giunta l'ora di compilare ed eseguire. Qui, devo aggiungere qualche dettaglio: nelle versioni relativamente più recenti del Go (dalla versione 1.11) la compilazione dei programmi passa attraverso i **Go Modules** che sono il sistema standard ufficiale per la gestione delle dipendenze nei progetti Go. Questo semplifica un po' la gestione rispetto al precedente sistema (che non è il caso di descrivere qui). In parole povere: prima di compilare bisogna preparare un file che si chiama **go.mod**, che normalmente sta nella stessa directory dove ci sono i sorgenti. Per crearlo ho scritto un piccolo **script-shell** (*buildmod.sh*) per facilitarmi (e facilitarvi) la vita: eccolo!

```
#!/bin/bash
rm go.mod
rm go.sum
rm mqttclient
go mod init mqttclient
go mod tidy
```

lanciando lo script vedrete qualcosa tipo questo:

```
aldo@Linux $ ./buildmod.sh
go: creating new go.mod: module mqttclient
go: to add module requirements and sums: go mod tidy
go: finding module for package github.com/eclipse/paho.mqtt.golang
go: found github.com/eclipse/paho.mqtt.golang in github.com/eclipse/paho.mqtt.golang v1.5.1
```

una volta creato il **go.mod** (solo per la prima compilazione, non serve ricrearlo ogni volta) si può compilare ed eseguire. Compiliamo (e questo è veramente semplicissimo):

```
aldo@Linux $ go build
```

ed eseguiamo in due terminali diversi (*possibilmente in quasi-contemporanea, per non perdersi messaggi: se siete troppo lenti, quando il secondo client parte il primo potrebbe già aver pubblicato tutti i suoi messaggi, ah ah ah. Comunque le "strane" sleep che ho aggiunto sono proprio per rendere leggibili i risultati*). Il risultato è questo:

TERMINALE 1:

```
aldo@Linux $ ./mqttclient Client2 test/topic1 test/topic2
Client2: mi sottoscrivo al topic test/topic2
messaggio ricevuto sul topic test/topic2: "messaggio 0 dal client Client1"
Messaggio 0 publicato sul topic test/topic1
messaggio ricevuto sul topic test/topic2: "messaggio 1 dal client Client1"
Messaggio 1 publicato sul topic test/topic1
messaggio ricevuto sul topic test/topic2: "messaggio 2 dal client Client1"
Messaggio 2 publicato sul topic test/topic1
```

TERMINALE 2:

```
aldo@Linux $ ./mqttclient Client1 test/topic2 test/topic1
Client1: mi sottoscrivo al topic test/topic1
Messaggio 0 publicato sul topic test/topic2
```

```
messaggio ricevuto sul topic test/topic1: "messaggio 0 dal client Client2"
Messaggio 1 publicato sul topic test/topic2
messaggio ricevuto sul topic test/topic1: "messaggio 1 dal client Client2"
Messaggio 2 publicato sul topic test/topic2
messaggio ricevuto sul topic test/topic1: "messaggio 2 dal client Client2"
```

Visto? Anche questa versione in Go funziona perfettamente (e identicamente) alle altre scritte in C!

Ok, per oggi può bastare. Nel prossimo articolo non so ancora di cosa parlerò, ma vi prometto che sarà, come sempre, molto interessante! Ah, come promesso all'inizio dell'articolo, dopo il solito saluto vi proporrò una citazione di Rob Pike sulla genesi del Go. Leggetevi la citazione (e magari anche **l'intero articolo di Pike**) è molto interessante!

Ciao, e al prossimo post!

"...Intorno a settembre 2007, stavo lavorando a una modifica minore ma fondamentale su un enorme programma C++ di Google, uno con cui tutti voi avete avuto a che fare, e le mie compilazioni impiegavano circa 45 minuti sul nostro gigantesco cluster di compilazione distribuito. Fu annunciato che ci sarebbe stata una presentazione tenuta da un paio di dipendenti di Google membri del comitato per gli standard C++. Avrebbero illustrato le novità in arrivo con il C++0x, come veniva chiamato all'epoca. (Oggi è noto come C++11). Nel corso di quell'ora di intervento abbiamo sentito parlare di circa 35 nuove funzionalità che erano in fase di progettazione. In realtà ce n'erano molte di più, ma solo 35 sono state descritte durante l'intervento. Alcune delle funzionalità erano minori, ovviamente, ma quelle citate nella presentazione erano almeno abbastanza significative da meritare di essere menzionate. Alcune erano molto sottili e difficili da comprendere, come i riferimenti rvalue, mentre altre sono particolarmente tipiche del C++, come i modelli variadici, e altre ancora sono semplicemente folli, come i letterali definiti dall'utente. A questo punto mi sono posto una

domanda: il comitato del C++ credeva davvero che il problema del C++ fosse che non avesse abbastanza funzionalità? Sicuramente, in una variante della battuta di Ron Hardin, sarebbe un risultato maggiore semplificare il linguaggio piuttosto che arricchirlo. Ovviamente è ridicolo, ma tenete a mente l'idea...” da “Less is exponentially more”, Rob Pike, 2012

https://italiancoders.it/mqtt-come-usare-mqtt-in-c-pt-3/