



JULIA: Verifica di Software Java e Android



Spin-off dell'Università degli Studi di Verona
<http://www.juliasoft.com>

Sommario

JULIA è un sofisticato strumento software che identifica, in modo del tutto automatico, bug in programmi Java e Android, prima che questi bug vengano eseguiti e possano quindi avere conseguenze negative. L'identificazione dei bug avviene tramite le più moderne tecniche di interpretazione astratta, in maniera del tutto autonoma dall'utente e senza richiedere il codice sorgente del programma analizzato ma solo il .jar del codice oggetto. Il sistema JULIA è utilizzabile da linea di comando o da web e permette di analizzare in pochi minuti programmi Java e Android di dimensione significativa. Il risultato è una breve lista di potenziali bug, con la garanzia formale che essa include tutti i veri bug del programma.

1. L'Idea



La complessità del software è ormai così elevata che viene data per scontata la presenza di errori o bug nei programmi prodotti e venduti anche dalle più blasonate società mondiali di software. Questi errori si manifestano dopo che il software è stato venduto, provocando, nei casi più fortunati, la perdita di dati e tempo oppure, nei casi peggiori, conseguenze più gravi se l'errore blocca apparecchiature telefoniche, medico/operatorie, industriali o siti web. La correzione dei bug diventa quindi una delle attività più difficili e costose che le società di software devono affrontare, spesso dopo che il prodotto software è già stato venduto e usato, con conseguenze anche sull'immagine stessa della società che ha prodotto il software errato.

La ricerca automatica dei bug è quindi un'attività di interesse per le società che producono software, sia nei casi in cui tale produzione è il loro core business, sia nei casi in cui la società si occupa di altre attività e ha una limitata produzione di software. In tal senso, è possibile ridurre la presenza di bug utilizzando:

- linguaggi di programmazione evoluti, con tipi forti e controlli statici a tempo di compilazione: la stessa compilabilità del programma è in questo caso una garanzia parziale della sua correttezza. Linguaggi come Java e i suoi dialetti, tipo Android, sono i più tipici rappresentanti di questa categoria di linguaggi che abbiano avuto un buon successo commerciale;
- strumenti automatici che cercano quei bug che inevitabilmente superano i controlli statici dei compilatori: errori di accesso al riferimento null, non terminazione del codice, esaurimento delle risorse, risultati numerici troppo approssimati o overflow.

JULIA è uno di questi strumenti automatici. A differenza dei suoi pochi concorrenti, JULIA fornisce una garanzia di correttezza: i bug del programma analizzato sono sicuramente nella lista di bug potenziali che JULIA ha identificato e questo è vero anche per quei programmi che utilizzano strutture dati dinamiche in memoria, condivise o cicliche. Come conseguenza, è sufficiente verificare la lista di bug fornita da JULIA e convincersi che essi siano dei falsi allarmi, per concludere che il software è corretto. Questo non è vero per gli altri prodotti concorrenti o è vero con un livello di precisione molto più basso o limitatamente ai programmi che usano esclusivamente variabili di tipo primitivo o che sono scritti secondo un'opportuna disciplina. JULIA si prende cura inoltre di fornire una lista ristretta di potenziali bug, in modo da limitare il tempo necessario per la loro verifica. Per esempio, la precisione di JULIA per l'analisi di accesso al riferimento null è tra il 98% e il 100%, contro un 80% del più diretto strumento concorrente sempre corretto.

2. Il Software JULIA



Il software JULIA è stato sviluppato da ricercatori dell'Università di Verona. Essi hanno deciso di concretizzare le loro avanzate conoscenze scientifiche, conseguenti alla loro attività di ricerca, in uno strumento software che possa essere preciso ed efficiente, quindi utile al mondo industriale. Il risultato è utilizzabile via internet connettendosi al sito <http://www.juliasoft.com>, senza bisogno di installarlo localmente sulla propria macchina. È sufficiente fornire uno o più archivi .jar contenenti il codice Java o Android compilato, che si intende analizzare, e selezionare l'analisi che si desidera. Il sistema remoto fornirà il risultato dopo alcuni minuti di lavoro.

La scelta di uno strumento web ha il vantaggio che il sistema è continuamente tenuto aggiornato: i più recenti risultati teorici della ricerca scientifica e l'ultima versione del software sono immediatamente disponibili agli utenti in rete. Inoltre questo riduce i costi di installazione e manutenzione del programma da parte dei suoi utilizzatori e non richiede l'acquisto di alcuna licenza.

3. Trova Più Bug che FindBugs

JULIA ha automaticamente identificato bug di riferimento null nell'editor Java EJE. JULIA ha identificato loop infiniti e quindi non terminanti, errati confronti fra stringhe e bug di riferimento null in quattro esempi di software inclusi nella distribuzione di Android della Google e in due applicazioni Android a codice aperto. Abbiamo analizzato gli stessi programmi con FINDBUGS, il tool di verifica statica più scaricato al mondo, usato anche da grandi multinazionali del software. Esso considera i tipi di bug che JULIA ha trovato, quindi il confronto è possibile. Il risultato è impressionante: FINDBUGS è più veloce di JULIA, ma non segnala nessuno dei bug e dichiara i programmi analizzati come privi di bug. Ecco la superiorità di un'analisi semantica come quella di JULIA, basata su metodi formali, rispetto a un'analisi sintattica come quella di FINDBUGS.

4. La Società JULIA

La società JULIA nasce nel novembre 2010 come spin-off dell'Università di Verona al fine di trasferire su un piano industriale i risultati teorici della ricerca scientifica in ambito di analisi del software. Le sue attività sono:

- un servizio di analisi di software tramite un sito web, al momento gratuito ma in prospettiva a pagamento, con un prezzo proporzionale alla complessità dell'analisi richiesta e del software analizzato. Le analisi al momento disponibili includono quella del riferimento null, quella di terminazione, quella di class cast e quella del codice morto;
- un servizio di definizione e implementazione di nuove analisi di software Java e Android, su richiesta di terzi;
- sviluppo di software di qualità in Java, Android e Objective-C, usando gli strumenti di analisi e verifica della società.

Nel primo caso, JULIA si rivolge a società e individui che producono software e vogliono migliorarne la qualità e ottenere una dimostrazione formale di alcuni aspetti della sua correttezza. Nel secondo e terzo caso, essa si rivolge a società e individui che operano nell'ambito dei linguaggi di programmazione e della programmazione in condizioni critiche (telefonia, medicina, software industriale o militare) e hanno quindi la necessità di verificare formalmente ed esaurientemente certe proprietà del software prima di renderlo disponibile agli utenti finali.

5. La Scelta di Java



JAVA è uno dei linguaggi maggiormente usati per la programmazione di software, sia di rete che locale. Il suo sistema di tipaggio forte e l'assenza di puntatori espliciti ne fanno un candidato ideale per lo sviluppo di analisi statiche corrette. L'indipendenza del bytecode Java dalla macchina che lo eseguirà fa sì che l'analisi di Java possa essere svolta a livello di codice bytecode compilato e rimanga valida indipendentemente dal computer sul quale il codice verrà poi eseguito. La presenza di una vastissima libreria standard permette inoltre di istruire JULIA sul comportamento di tale libreria e ottenere quindi analisi con un ottimo livello di precisione.

6. La Scelta di Android



ANDROID è un dialetto di Java sviluppato da Google per la programmazione di applicazioni telefoniche e sistemi embedded. Condivide la sintassi e la semantica di Java e parte delle sue librerie, pur aggiungendone altre. Ridefinisce il bytecode Java per permettere una riduzione della dimensione del codice compilato. La compilazione di una applicazione Android passa comunque dalla generazione temporanea di classi Java compilate in Java bytecode. È quindi possibile analizzare le applicazioni Android in tale formato, prima di tradurle nel loro bytecode ottimizzato. L'uso di applicazioni Android in contesti sensibili alla presenza di bug o alla violazione della privacy, come è il caso delle applicazioni per telefoni cellulari, rende significativa l'analisi statica di applicazioni Android.

7. Conclusioni

Il progetto JULIA si presenta come una realtà scientifica e imprenditoriale avanzata e visionaria. Il suo elemento caratterizzante è la scommessa su un servizio di analisi di software remoto via web, che garantisce il costante aggiornamento del sistema e non richiede l'acquisto di licenze del software, benché la società operi anche tramite canali più tradizionali come la vendita di licenze e lo sviluppo software. La complessità del problema dell'analisi statica di codice reale e i risultati positivi già ottenuti rendono JULIA una società unica e un esempio di imprenditoria centrata sulla ricerca universitaria e da essa direttamente derivata.

8. Chi Siamo



La società è presieduta da Fausto Spoto, dell'Università di Verona (fausto.spoto@univr.it). Gli altri soci sono il professor Roberto Giacobazzi dell'università di Verona e due imprenditori: Paolo Fiorini di M&A Partners Srl e Paolo Errico di Maxfone Srl.