Commit Risk Assessment for CDH Components

Wiandt Bernát
2018. 03. 15.

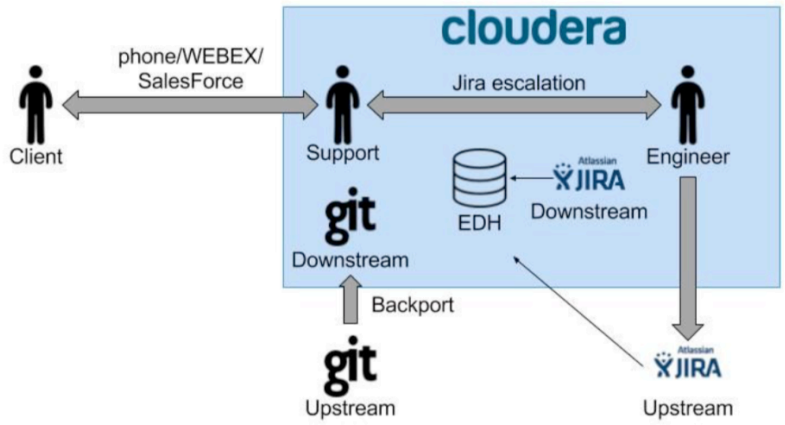
Bevezető

A Cloudera cég fő terméke a CDH disztribúció, amely nagy adatmennyiségeket hatékonyan kezelő open source szoftverkomponenseket (big data eszközök) integrál egy egységes egészbe. Ehhez az open source szoftvereket kis mértékben módosítják, azokból saját belső verziót tartanak fent. A fejlesztés során sűrűn előforduló feladat, hogy a cég által karbantartott belső szoftververzióba (downstream) a publikus (upstream) forrásból át kell emelni kódváltoztatásokat (commit). A művelet során akaratlanul hibákat okozó kódrészletek is bekerülnek a privát (downstream) verzióba.

A PARIPA programon belül végzett kutatómunka célja, hogy a Cloudera cégen belül és upstream történő fejlesztés során előálló commitok veszélyességét képesek legyünk automatizáltan meghatározni. A veszélyességet, mint mérőszámot a fejlesztői munkafolyamat támogatására lehet majd hatékonyan felhasználni. A munka motivációja a következő: a CDH disztribúción belül vannak kritikus komponensek, amikre sok másik szoftver funkcionalitása épül. Ezek módosítása, akár hibajavítás céljával is, kockázatos.

Az első félévben a rendszer megértése a cél, majd az egyes commitok jellemzése különböző tulajdonságok alapján: szerző, dátum, méret, hány komponenst módosít, milyen jellegű változtatás. Utóbbi tulajdonság meghatározásához szükséges a cég által használt JIRA rendszer és a commitokat kezelő git között egy kapcsolat létrehozása, amely lehetőséget teremt az egyes kódváltoztatások besorolására.

A CDH disztribúció fejlesztési folyamata



A fenti ábrán a Cloudera fejlesztési folyamatának kivonata látható. A CDH disztribúcióban használt Hadoop és más Apache projektek fejlesztése egy nyílt forráskódú git repository-ban történik, ezt az ábrán az Upstream Git-tel jelöltük. A Downstream git repository-ban van eltárolva a CDH disztribúcióhoz tartozó forráskód, ez a forráskód már nem érhető el bárki számára, csak azon személyeknek akiknek erre a cég engedélyt adott. A két Git repository között nagy hasonlóság van, hisz ugyanazon komponensekhez fejlesztése történik meg bennük.

Általánosságban elmondható, hogy a fejlesztések nagy többsége az Upstream Git repository-ban történik, ahonnan később a fontosabb kódváltoztatásokat backportolják (emelik át) a Downstream Git repository-ba.

Az ábráról még érdemes megemlíteni a Jira adatbázisokat. A Jira egy issue-tracking adatbázis, amiben szereplő adatok a Jira eszkalációk. Létezik egy upstream Jira minden egyes komponenshez, illetve létezik egy Downstream, ami csakis a CDH disztribúcióhoz tartozik.

A harmadik fontos elem az ábrán az EDH adatbázis. Ez az elem a projekt szempontjából is nagyon fontos, hiszen ezen keresztül értük el a downstream Jira issue-k adatait. Az EDH-ben ahogy az ábra is mutatja a Jira eszkalációk másolata kerül eltárolásra. Az adatbázis szerepe, hogy egy helyről visszakereshető legyen az összes Jira eszkaláció ami a CDH szempontjából fontos lehet.

Ha egy ügyfél problémát tapasztal a CDH disztribúcióval, vagy annak valamelyik komponensével. Akkor felveszi a kapcsolatot a Cloudera ügyfélszolgálatával.

Ekkor a cég egyik alkalmazottja segít a probléma megoldásában. Abban az esetben, ha nem tudja az ügyfélszolgálaton dolgozó alkalmazott megoldani az ügyfél problémáját, felveszi a kapcsolatot a cég egyik szoftvermérnökével. Ez az esetek nagy többségében egy Jira eszkalációt eredményez. Amely eszkaláció a Downstream Jira-ban kerül eltárolásra. Azon hibák amelyek megoldásához már egy szoftvermérnök tudása kell, általában bugoknak számítanak. Ha a tapasztalt hiba egy adott komponensre vonatkozik, akkor a mérnök megvizsgálja, hogy létezik-e megoldás az adott komponens nyílt forráskodú könyvtárában. Ha van létező megoldás akkor, az adott commit-ot backportolva, majd a kódváltoztatás letesztelése után a felhasználó rendelkezésére bocsátja.

Abban az esetben ha nem létezik olyan commit, amely megoldja a felhasználó problémáját, akkor a feladat megoldásával megbízott mérnök létrehoz egy Jira ticket-et az adott komponens upstream Jira adatbázisában. Így a komponensen dolgozó mérnököknek (nem feltétlenül csak a Cloudera mérnökei) már van információja arról, hogy milyen problémára, bug-ra kell megoldást találni. A hiba kijavítása után az upstream Git repository-ból backportolva a megfelelő commitot kerül a kódváltoztatás a CDH git repository-ba.

Habár a backportolt commitok általában egy bugot javítanak ki a CDH valamelyik komponensében, gyakran tartalmaznak bugokat amelyek csak később fognak problémát okozni. Az előbbi leírás alapján egyértelmű, hogy a Jira eszkalációk száma és a commitok között valamilyen korrelációnak kell lennie.

A projekt célja, egy olyan rendszer elkészítése amely képes az egyes commitokhoz egy olyan kockázati értéket rendelni amely alapján eldönthető, hogy mely commitokat érdemes jobban letesztelni.

Commitok jellemzése

Ahhoz, hogy az egyes commitokkal járó kockácatot meg tudjuk becsülni, először egy a coomitok jellemzésére alkalmas ún. jellemző (feature) halmazt kell definiálnunk. A feature-ök meghatározására a Gitben rendelkezésre álló információkat használtuk fel. Több tervezett feature-t is megállapítottunk, amelyek a félév során nem kerültek megvalósításra.

Ilyen tervezett feature például:

- A commit szerzője (author)
- commitot beküldő személy (commiter)
- commiter, author személyének egyezése
- a commit létrejöttének dátuma (commit date)
- a változtatott fájlok száma
- a commit mérete
- a git diff (különbség két commit között)
- módosított régiók száma (itt a régió forráskódon belüli, összefüggő részeket jelent)

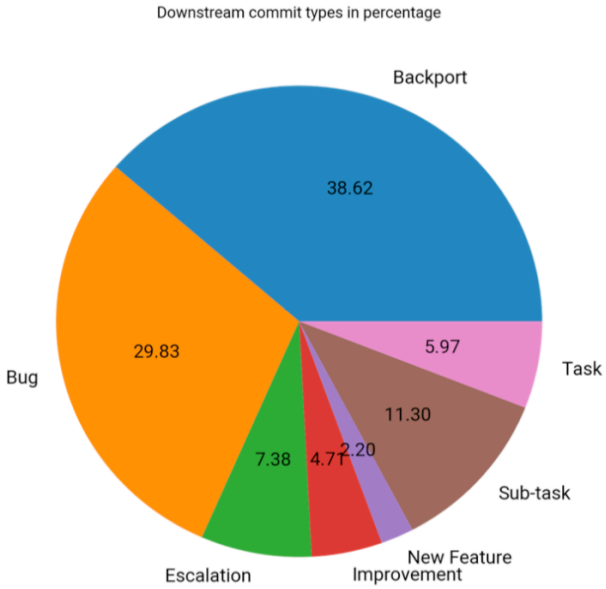
Az egyetlen jellemő, aminek megvalósítása megtörtént a félév során a commit típus (commit type) volt. A commit type jelentése nem más mint az adott commitot létrehozó Jira eszkaláció típusa.

A commit type feature megállapításához párosítani kell a commithoz tartozó Jira hibajegyet. A Git commit message segítségével megtalálható a commithoz tartozó Jira hibajegy. Ezek a commit message-ek néhány egyszerű mintát követnek. Backport típusú commitok esetében például:

- “ CDH-62517 HIVE-16890:...” vagy
- “CDH-61955: Backport HIVE-17969:.....”

Cloudera belső fejlesztői commitok esetén a commit message-ben a “CLOUDERA-BUILD” kifejezés majd a Jira hibajegy azonosítója szerepel.

Ennél a két kategóriánál egyszerűen meghatározható a Jira issuekey a commit üzenetből. Néhány commit esetén a teljes commit message megegyezik a Jira hibajegy címével, így létrehozható köztük a kapcsolat. A commit üzenetet felhasználva tehát létrehozható a kapcsolat a belső Git repository-ban található commitok és a Jira hibajegyek között.



Ezt a kapcsolatot kihasználva tudtam megállapítani, hogy az egyes commitoknak mi a jira típusa. A kapott eredményeket egy tortadiagrammon ábrázolva könnyen belátható, hogy a bug és a backport commitok a leggyakoribbak. Ami részben alátámasztja az elméletet miszerint a backport commitokkal gyakran kerül hiba a CDH forráskódjába.

További célok

A projekt további folytatásában, a következő félév is szeretnék részt venni. Több olyan ötlet felmerült a Témalaboratórium keretein belül amelyek megvalósítása hasznosnak bizonyulhat a későbbiek során.

Ezek az ötletek:

- Több feature hozzáadása az adatbázishoz a Git-ből illetve újabb feature-ök gyűjtése a Jira adatbázisból is
- Több CDH komponensre kiterjeszteni az adatbázist (például: Pig, Impala)
- Esetleges al-komponensek definiálása (például: Metastore, Hcatalog)
- A SalesForce ticket-ek felhasználása a számításokban
- Ruzsbánszki Nándor projektjének (CDH usage pattern analysis) eredménynek bevétele a számításba