

Arttu Ylä-Sahra

**Katsaus NPM-pakettivaraston metatietoihin ja pakettien
riippuvuussuhteisiin**

Tietotekniikan pro gradu-tutkielma

7. kesäkuuta 2020

Jyväskylän yliopisto

Informaatioteknologian tiedekunta

Tekijä: Arttu Ylä-Sahra

Yhteystiedot: eearyla@student.jyu.fi

Ohjaaja: Antti Valmari

Työn nimi: Katsaus NPM-pakettivaraston metatietoihin ja pakettien riippuvuussuhteisiin

Title in English: Review of metadata and package dependency relations in the NPM repository

Työ: Pro gradu-tutkielma

Opintosuunta: Tietotekniikka

Sivumäärä: 39+24

Tiivistelmä: Pakettivarastot ovat nykyajan ohjelmoinnissa hyvin yleisiä sovelluskehityksen välineitä, ja NPM (Node Package Manager) on yksi merkittävimmistä pakettivarastoista. Tässä tutkielmassa tehdään poikittainen katsaus NPM:n metadatasisältöön loppuvuonna 2019. Tutkielmassa esitellään keräystapa tarvittaville metatiedoille, ja koostetaan tilastoja yleisellä tasolla sekä tutkitaan pakettien keskinäisiä riippuvuussuhteita. Tutkimuksen yhteydessä havaittiin pakettien riippuvuuksien merkittävää jakautumista suosion mukaan. Tutkimuksessa havaittiin myös kehittyvien pakettien nopeaa päivittymistä, kuitenkin huomioiden että enemmistöllä paketeista on vain muutama versio.

Avainsanat: Node Package Manager, metadata, riippuvuussuhteet, tilastointi, ohjelmapaketit, paketinhallinta, ohjelmistokehitys

Abstract: Package repositories are one of the most common tools of software development today, and NPM is one of the most significant package repositories. In this thesis, a cross-sectional study of the NPM metadata at the end of year 2019 is completed. A method for collecting data is presented, and statistics collated on both a general level and interpackage dependency relationships are studied. It was found that, among other findings, the dependencies used are sharply divided between popular and less popular packages. It was also observed that developing packages tend to update quickly, with the caveat that a majority of

packages do not have more than few versions.

Keywords: Node Package Manager, metadata, dependency relationships, statistics, software packages, package management, software development

Termiluettelo

| | |
|------------------------|--|
| CouchDB | Apache Foundationin kehittämä, kyselyiltään JavaScript-kieleen ja tallennusformaatiltaan JSON:iin perustuva dokumenttitietokanta. Kts. https://couchdb.apache.org/ |
| GraphViz | AT&T Labs Researchin kehittämä ohjelma DOT-kielellä kuvailtujen graafien visualisointiin. |
| JSON | <i>JavaScript Object Notation</i> , JavaScript-kielen oliomalliin perustuva avoin, tekstimuotoinen kuvauskieli |
| Node.js | JavaScript-ajoympäristö. Kts. https://nodejs.org/ |
| NPM | <i>Node Package Manager</i> , Node.js-ympäristössä käytetty pakettivarasto. Kts. https://www.npmjs.com/ |
| Pakettivarasto | Engl. <i>package repository</i> , ohjelmapakettien koodin ja metatietojen (esim. julkaisupäivämäärä, pakettia kuvaavat avainsanat) säilytyspaikka |
| PL/PGSQL | Postgres-tietokannassa käytettävä proseduraalinen ohjelmointikieli. |
| Ruby | Dynaaminen, olio-orientoitunut avoimen lähdekoodin ohjelmointikieli. Kts. https://www.ruby-lang.org |
| Rust | Staatteisesti tyypitetty, erityisesti muisti- ja säieturvallisuuteen sekä nopeuteen keskittyvä ohjelmointikieli. Kts. https://www.rust-lang.org/ |
| Semanttinen versiointi | Versiointikäytäntö, jossa eri versiotunnisteen osat merkitsevät erityyppisiä muutoksia edelliseen versioon nähden. Kts. https://semver.org/ |
| SSD | <i>Solid state drive</i> , tietokoneessa käytettävä massamuisti jossa ei ole liikkuvia osia |
| Versiomääre | Yhteensopivien pakettiversioiden kuvaus riippuvuuksien yhteydessä |
| Versiotunniste | Yksittäisen version yksilöivä tunniste. Monesti merkkijono, esim. semanttisen versioinnin mukainen 1.12.10.alpha-1 |

Kuviot

| | |
|--|----|
| Kuvio 1. Raakadatan maksimikoko kvantiileittain..... | 4 |
| Kuvio 2. Pakettiversioiden selviytymisennuste 12 kuukauteen asti..... | 19 |
| Kuvio 3. Pakettiversioiden selviytymisennuste 90 päivään asti | 20 |
| Kuvio 4. Riippuvuusmäärien kvantiilit 0.31-1 | 21 |
| Kuvio 5. Uusien pakettiversioiden määrä lyhyimmän askeletäisyyden mukaan <code>lodash@4.17.5-</code> paketille | 22 |
| Kuvio 6. Uusien pakettiversioiden määrä lyhyimmän askeletäisyyden mukaan <code>left-pad@1.3.0-</code> paketille | 25 |
| Kuvio 7. Uusien pakettiversioiden määrä lyhyimmän askeletäisyyden mukaan 56% riippuvuuskvantiilille | 26 |

Taulukot

| | |
|---|----|
| Taulukko 1. Luettelo NPM-rekisterikyselyn pakettikohtaisista avaimista ja niiden si- sällöstä..... | 7 |
| Taulukko 2. Luettelo NPM-rekisterikyselyn pakettiversiokohtaisista avaimista ja nii- den sisällöstä | 8 |
| Taulukko 3. Kuvaus erilaisista pakettimetatiedoissa havaituista epäsäännöllisyyksistä | 12 |
| Taulukko 4. <code>lodash@4.17.5</code> -pakettiversion riippuvuusverkon riippuvien pakettien määräjakauma askelittain..... | 23 |
| Taulukko 5. <code>left-pad@1.3.0</code> -pakettiversion riippuvuusverkon riippuvien paket- tien määräjakauma askelittain | 25 |
| Taulukko 6. 56% riippuvuuskvantiilin riippuvuusverkon riippuvien pakettien määrä- jakauma askelittain..... | 27 |

Sisältö

| | | |
|-------|---|----|
| 1 | JOHDANTO | 1 |
| 1.1 | Yleistä..... | 1 |
| 1.2 | Tästä tutkimuksesta | 2 |
| 1.3 | Tutkielman rakenne | 2 |
| 2 | TIEDONKERUU JA TIETOLÄHTEET..... | 3 |
| 2.1 | Yleistä..... | 3 |
| 2.2 | Rekisterin metatietojen keräys | 3 |
| 2.2.1 | Hakumenetelmä ja rakenteellistaminen..... | 3 |
| 2.2.2 | Kerätyt tietotyypit | 6 |
| 2.2.3 | Tiedon käyttörajoitukset | 8 |
| 3 | TIEDON JÄSENTÄMINEN JA TUTKIMUSMETODOLOGIA | 10 |
| 3.1 | Yleistä..... | 10 |
| 3.2 | Normalisointi ja kohtuullisen työmäärän rajaaminen..... | 10 |
| 3.3 | Versiot ja versioaikasarjat | 12 |
| 3.3.1 | Semanttinen versiointi | 12 |
| 3.3.2 | Versioaikasarjojen muodostus ja päivitystahti..... | 13 |
| 3.4 | Riippuvuuksien selvittäminen | 14 |
| 3.5 | Graafinmuodostus ja -analyysi..... | 15 |
| 4 | TULOKSET..... | 16 |
| 4.1 | Tilastojen tulkinnasta..... | 16 |
| 4.2 | Yleiset pakettitilastot | 16 |
| 4.2.1 | Lisenssit | 16 |
| 4.2.2 | Suosikkimerkinnät | 17 |
| 4.2.3 | Avainsanat..... | 17 |
| 4.2.4 | Nimikkeet | 17 |
| 4.2.5 | Pakettien versiomäärät..... | 18 |
| 4.3 | Pakettiversioiden ominaisuuksia..... | 18 |
| 4.3.1 | Versioiden päivittyvyys..... | 18 |
| 4.3.2 | Versiotunnisteet | 19 |
| 4.3.3 | Riippuvuuksien versiomääreet | 19 |
| 4.4 | Pakettien riippuvuussuhteet | 21 |
| 4.4.1 | lodash@4.17.5 | 22 |
| 4.4.2 | left-pad@1.3.0..... | 24 |
| 4.4.3 | 56% kvantiili | 26 |
| 5 | YHTEENVETO..... | 28 |
| 5.1 | Johtopäätöksiä | 28 |
| 5.2 | Jatkotutkimusideoita | 28 |
| 5.2.1 | Visualisointi | 29 |
| 5.2.2 | Metatiedot ja koodi | 29 |

| | |
|--|----|
| LÄHTEET | 30 |
| LIITTEET..... | 33 |
| A SQL-tietokantarakenteen skeema pääpiirteittäin..... | 33 |
| B Versioaikasarjojen luontialgoritmi PL/PGSQL-kielillä | 40 |
| C Esimerkki Kaplan-Meier-estimaattorista PL/PGSQL-kielillä | 46 |
| D 100 yleisintä paketissa tavattua lisenssiä | 49 |
| E 100 suosituinta pakettia suosikkimerkintöjen perusteella | 51 |
| F 100 suosituinta avainsanaa pakettiesiintymien perusteella | 53 |
| G 100 eniten riiputuinta pakettiversiota..... | 55 |

1 Johdanto

1.1 Yleistä

Useilla ohjelmointikielillä on nykyään vähintään yksi vakiintunut yleinen pakettivarasto. JavaScript on yksi suosituimmista ohjelmointikielistä (Carbonnelle 2020), ja sen kaikkein suosituin yleinen pakettivarasto on NPM. NPM keskittyy pääosin Node.js-ajoympäristössä käytettäviin paketteihin. Tämä ei kuitenkaan rajoita pakettien sisältöä pelkästään Node.js-ympäristössä suoritettavaan koodiin, vaan on yleistä myös hakea esim. verkkoselaimessa suoritettavaa sisältöä NPM-pakettivarastosta, ja paketoita (engl. *bundle*) sisältö käyttötarkoituksen mukaiseen muotoon.

Pakettivarastojen ja riippuvuuksien tutkiminen ei ole millään tavalla uutta. Cox pohtii riippuvuuksiin liittyviä vaaroja, nostaen esille erilaisia riskejä mitä koodin uudelleenkäyttöön voi liittyä (2019). Decan, Mens ja Grosjean kuvaavat tutkimuksessaan vertailevasti seitsemän eri pakettivaraston muutamia eri peruslinalaisuuksia. Tutkimus asetti muutamia tutkimuskysymyksiä, jotka ovat hyödyllisiä myös tämän tutkimuksen kannalta:

- Kuinka usein paketit päivittyvät?
- Missä määrin paketit riippuvat toisistaan?
- Kuinka yleisiä transitiiviset (epäsuorat) riippuvuudet ovat?

(Decan, Mens ja Grosjean 2019, s. 389-401).

Jyväskylän yliopistossa aihetta aiemmin tutkinut Muranen perehtyi pro gradu-tutkielmassaan erityisesti pakettien turvalliseen hallintaan. Tutkimus ei keskittynyt vain NPM-pakettivarastoon, vaan tutustui myös muiden yleisten pakettivarastojen toimintaan ja käytänteisiin turvallisuusnäkökulmasta. (Muranen 2019).

Myös erityisesti NPM:ään kohdistuvaa tutkimusta on tehty aiemmin, monista eri näkökulmista. Abdalkareem ym. (2017) ovat tutkineet erityisesti pienten NPM-riippuvuuksien käytötapoja kehittäjäkyselyllä. Teknistä (päivittymis)viivettä NPM-pakettien riippuvuuksien suhteen (engl. *technical lag*) on myös tutkittu erikseen, ja muodostettu metriikka tämän kvan-

tifioimiseen (Zerouali ym. 2019). Lisäksi Cox ym. ovat kehittäneet yleisempää metriikkaa riippuvuuksien tuoreuden mittaamiseen (2015).

1.2 Tästä tutkimuksesta

Tämän tutkimuksen tarkoituksena on toteuttaa kattava, poikittaisluonteinen katsaus NPM-pakettivaraston nykytilaan. Tavoitteena on selvittää sekä yleisluontoisempia tilastollisia tietoja (esim. mitä lisenssejä käytetään eniten) että perehtyä tarkemmin pakettien välisiin suhteisiin (esim. minkätyyppisiä riippuvuuksia paketeilla on). Aika- ja resurssirajoitteista johtuen tutkimus toteutetaan suhteellisen yleisluontoisella tasolla, eikä kokonaan uusia metriikoita kehitetä.

1.3 Tutkielman rakenne

Luvussa 2 selostetaan tiedonkeruun peruseriaatteen ja käytetyt tietolähteet. Luvussa 3 tarkastellaan tarkemmin tiedon jalostamis- ja tutkimusmenetelmiä, sekä tutkimuksen eri väli vaiheita kokonaisuuksina. Luvussa 4 kuvaillaan tutkimuksen tuloksia tiiviisti ja tehdään havaintoja tulosten luonteesta. Luvussa 5 tehdään johtopäätös tutkimuksesta, ja pohditaan myös mahdollisia jatkotutkimuksen aiheita.

2 Tiedonkeruu ja tietolähteet

2.1 Yleistä

Pääasiallisena tietolähteenä on käytetty NPM-pakettivaraston rekisteriä, joka on saatavilla osoitteesta <https://registry.npmjs.org>. Tiedonkeräys sijoittui pääosin jaksolle 20.10.2019–3.12.2019, edistyen keskimäärin noin 25 749 paketin päivätahdilla. Pitkähkö keruu-aika johtui pääasiassa tiedon suurehkosta määrästä yhdistettynä rajoitettuun keräysnopeuteen – keräyksen valmistuttua raakadatan koko oli pakattuna yli 8 gigatavua, ja pakkaamattomana noin 30 gigatavua.

Kuvio 1 kuvaa pakettien kokojakaumaa. Pakettien metadatan koon muutos vaikuttaisi melko tasaiselta logaritmisella skaalalla, lukuunottamatta aivan ääripäässä tapahtuvaa kasvua.

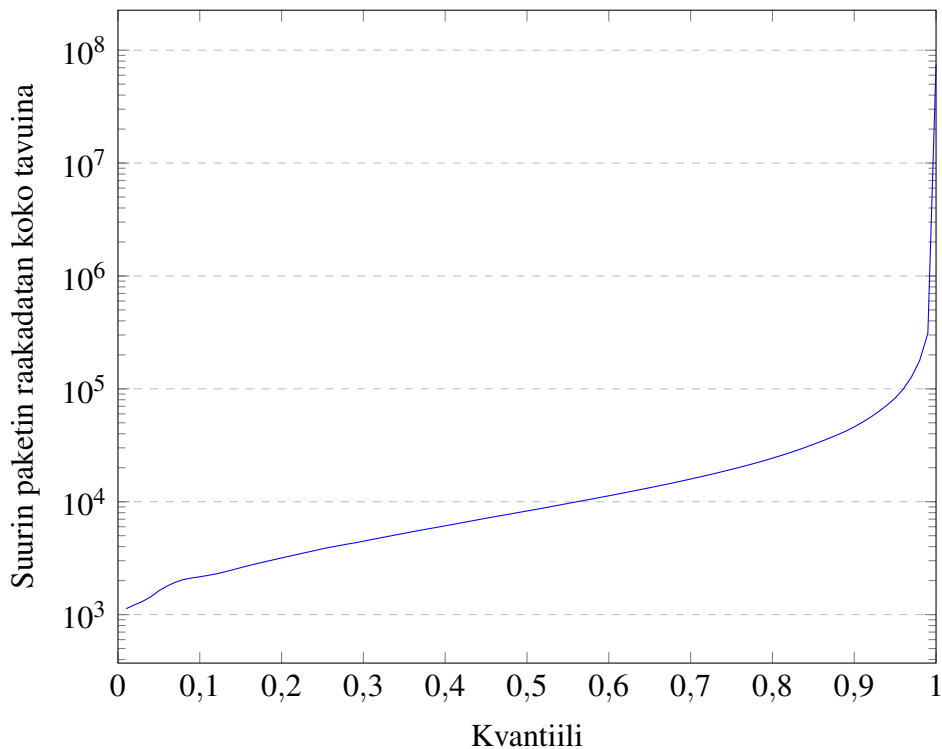
Lisäksi 3505 paketin tiedot päivitettiin tämän keruuajan jälkeen, sillä osa tiedoista oli jäänyt keräämättä käytetyn verkkotiedostojärjestelmän kirjainkoko-ominaisuuksien vuoksi. Käytetty tiedostojärjestelmä säilytti kirjainkoon, mutta salli vain yhden yhdistelmän samanaikaisesti, johtaen ongelmiin muuten samannimisten pakettien kanssa (esim. `jquery` ja `jQuery`).

Tällä menetelmällä on saatu mahdollisimman ajantasaista tietoa. Valmiiksi koostettuja lähteitä on kyllä olemassa (esim. Libraries.io:n tietokanta) (“Open Data - Libraries.io” 2020), mutta niistä löytyvä tieto osoittautui alustavan tarkastelun yhteydessä liian vanhaksi ja hankalasti jäsennettäväksi verrattuna suoriin NPM-rekisterikyselyihin. Libraries.io:n tietokanta oli keräyksen aloitushetkellä päivitetty edellisen kerran vuonna 2018 (versio 1.2.0), joka oli tutkimuksen tarkoitusta varten aivan liian pitkä aika.

2.2 Rekisterin metatietojen keräys

2.2.1 Hakumenetelmä ja rakenteellistaminen

NPM-rekisteristä pyydettiin tietoja pakettikohtaisilla HTTP-pyyntöillä, esim. pyytämällä osoitetta <https://registry.npmjs.org/react> `react`-paketille. Olemassaolevien pakettien nimiluettelo haettiin erikseen NPM:n julkiselta tietokantapalvelimelta (Amery 2018),



Kuvio 1: Raakadatan maksimikoko kvantiileittain

käyttäen dokumentoitua CouchDB-rajapintaa (Apache Foundation 2020). Tiedonkeruu on tietyin rajoituksin NPM:n avoimien lisenssiehtojen valtuuttamaa toimintaa (npm, Inc. 2019a) – tarkempiin yksityiskohtiin palataan luvussa 2.2.3.

Keruuolosuhteita varten kehitettiin erillinen Ruby-ohjelma, joka suoritti annetun nimiluettelotiedoston mukaisesti kyselyjä, ja tallensi tuloksen sekä kyselyn päivämäärän erillisiksi JSON-muotoisiksi tiedostoiksi. Ohjelman toimintaperiaate suunniteltiin hyvin yksinkertaiseksi: lue nimiluettelosta rivi, tarkista onko kyseistä pakettia jo ladattu, ja lähetä pyyntö NPM-rekisteriin. Lisäksi ohjelma kykeni yksinkertaisella tasolla analysoimaan pakettiriippuvuuksia, joka osoittautui tarpeelliseksi rekisterin nimilistan osoittauduttua tuntemattomasta syystä epätäydelliseksi.

Ohjelma lisäksi ilmoitti tarkoituksensa `User-Agent`-otsakkeessa esittäytymällä nimellä `“fetch-npm-list(thesis research tool, contact <eearyla@student.jyu.fi> for futher information)”`, kuten hyvään käytötapaan kuuluu.

Keräys toteutettiin asentamalla ohjelma Raspberry Pi-minitietokoneeseen, tallentaen tuloksensa ulkoiselle verkkolevyille. Tämä järjestely salli kyselyiden tekemisen automatisoidusti vuorokauden ympäri, mikä epäilemättä nopeutti tiedonkeräystä. Toisaalta selvittämättömmiksi jääneet tekniset ongelmat aiheuttivat Raspberry Pi:n kaatumisen melko säännöllisin väliajoin, vaatien joka kerta käsin tehtävää uudelleenkäynnistystä ja tallennettujen tietojen tarkistusta.

Keräämisprosessin ja varmuuskopioiden tekemisen jälkeen kerätty tieto vietiin rakenteiseen SQL-tietokantaan. Tietokanta oli aluksi SQLite-tyyppinen, mutta tietokannan teknisten rajoitteiden tultua vastaan otettiin käyttöön PostgreSQL-tietokanta Linux-pohjaisella palvelimella. Tällöin data siirrettiin SQLite-tiedostosta käyttäen `pg_loader`-ohjelmaa. Tarkempi tietokantaskeema on selostettu liitteessä A.

2.2.2 Kerätyt tietotyypit

Yksittäinen pakettikysely palauttaa seuraavassa taulukossa esitettyjä tietotyyppisiä (npm, Inc. 2019b, `responses/package-metadata.md`). Taulukkoon on listattu vain tutkimuksen kannalta merkittävät tietotyypit, jotka siirretään rakenteellistettuihin tietokantatauluihin – muut mahdolliset tiedot (esim. tarkat latausosoitteet eri versioiden koodille) jäävät raakadatan sisältävään tauluun.

| Avain | Selitys | Tarkempi kuvaus |
|------------------------|----------------------------------|---|
| <code>name</code> | Nimi | Paketin yksilöintiin käytetty uniikki nimi, esim. <code>react</code> . Nimen kirjainkoolla on merkitystä. |
| <code>time</code> | Luonti- ja muokkautuspäivämäärä | Tieto siitä milloin paketti on ensimmäisen kerran luotu rekisteriin, ja milloin sitä on edellisen kerran muokattu. Sisältää lisäksi versiokohtaiset julkaisupäivämäärät. |
| <code>users</code> | Paketin merkinneet NPM-käyttäjät | Tieto kaikista NPM-käyttäjistä, jotka ovat merkinneet suosikiksi (engl. <i>favorite</i>) tämän paketin. |
| <code>dist-tags</code> | Paketin nimikkeet | Paketin eri versioihin kohdistetut nimikkeet (engl. <i>tag</i>). Jokaisella paketilla on aina <code>latest</code> -nimike, joka ilman muuta erillistä valintaa osoittaa aina paketin viimeisimpään versioon. |
| <code>versions</code> | Paketin versiokohtaiset tiedot | Semanttisen versionumeron mukaan järjestetyt tiedot paketin senhetkisestä tilasta. Osa viimeisimmän version tiedoista nostetaan osaksi paketin yleisiä tietoja, mutta niitä ei mainita tässä taulukossa erikseen. |
| <code>keywords</code> | Paketin avainsanat | Pakettia vapaamuotoisesti kuvaavat avainsanat. |

Taulukko 1: Luettelo NPM-rekisterikyselyn pakettikohtaisista avaimista ja niiden sisällöstä

Yksittäisestä paketin versiosta tallennetaan seuraavat tiedot (npm, Inc. 2019b, responses/package-metadata.md).

| Avain | Selitys | Tarkempi kuvaus |
|-----------------------|---|---|
| deprecated | Paketin vanhentuminen | Tieto siitä onko paketti vanhentumassa tai poistumassa käytöstä, miksi, ja kuvaus mahdollisista jatkotoimenpiteistä. |
| dependencies | Paketin suorat riippuvuudet | Paketin vaatimat riippuvuudet, paketin nimestä ja versiomäärestä koostuvina pareina. |
| optional Dependencies | Paketin <i>valinnaiset</i> riippuvuudet | Sama kuin <i>dependencies</i> , mutta nämä riippuvuudet eivät ole välttämättömiä paketille (joskin usein suositeltavia). |
| dev Dependencies | Paketin <i>kehitysaikaiset</i> riippuvuudet | Sama kuin <i>dependencies</i> , mutta näitä riippuvuuksia tarvitaan vain <i>kehityksen</i> yhteydessä, ei normaalissa käytössä. Tällaisia ovat esim. koodityylin tarkastustyökälu, joita loppukäyttäjä ei tarvitse ohjelmaa käyttääkseen. |
| peer Dependencies | Paketin <i>vertaiset</i> riippuvuudet | Tieto minkä paketin kanssa tämä paketti on yhteensopiva, kuitenkin suoraan vaatimatta tämän riippuvuuden lataamista. Tämän tyyppisiä riippuvuuksia käyttävät esim. liitännäisohjelmat. |
| license | Paketin <i>SPDX-</i> lisenssitunniste | Tieto paketin lisenssistä standardisoidussa muodossa. |
| author | Paketin tekijä | Paketin tekijän nimi ja sähköpostiosoite. |
| homepage | Kotisivu | Paketin kotisivun URL-osoite. |
| bugs | Bugiraportointi | Paketin bugiraporttien URL-osoite. |

| | | |
|---------------------------|--------------------------------|---|
| <code>contributors</code> | Paketin luontiin osallistuneet | Paketin kehittämiseen osallistuneet henkilöt. |
| <code>maintainers</code> | Paketin ylläpitäjät | Pakettiin julkaisu- ja päivitysoikeudet omaavat henkilöt. |
| <code>repository</code> | Paketin repositorio | Paketin lähdekoodin ja muun materiaalin versionhallinta. |

Taulukko 2: Luettelo NPM-rekisterikyselyn pakettiversiokohtaisista avaimista ja niiden sisällöstä

2.2.3 Tiedon käyttörajoitukset

NPM:n avoimen käytön lisenssi asettaa joitakin rajoitteita kerätyn tiedon käytölle. Seuraavassa luettelossa esitetään tutkimuksen kannalta merkittävät rajoitteet. (npm, Inc. 2019a). Lisäksi hyvä tieteellinen käytäntö edellyttää huolellisuutta ja tarkkuutta tietoaineistojen käsittelyssä (*Hyvä tieteellinen käytäntö* 2012).

1. Ehdon pykälän nro. 4 mukaan pakettien tietoturvatietoa saa käyttää vain henkilökohtaisiin tai sisäisiin yritystoiminnallisiin tarkoituksiin. Muille ei saa tarjota kyseistä tietoa suoraan eikä osana muita tuotteita tai palveluita. Näistä syistä nimenomaisesti tietoturvaan liittyvät elementit rajataan tutkimuksesta pois, sillä tutkimus melkein välttämättä olisi edellyttänyt kyseisten tietojen julkaisua jossakin muodossa.
2. Hyväksyttävän käytön rajoitusten pykälän nro. 4 mukaan henkilötietojen monistaminen tai jakaminen ilman käyttäjien henkilökohtaista lupaa on kiellettyä. Joitakin kyselytyyppejä ei voinut suorittaa ilman että niistä vääjäämättä olisi ilmennyt jotain henkilötietoja (esim. nimimerkkejä tai nimiä, kotisivu- ja sähköpostiosoitteita), joten tietoa jalostettaessa pyrittiin mahdollisimman pian hävittämään tarpeettomat henkilötiedot tutkimusdatasta esim. pseudonymisoimalla ne. Tällöin avain pseudonyymeihin voitiin erottaa tietokannasta, myöhemmin toteutettavaa hävittämistä varten. Luonnollisesti näitä henkilötietoja ei myöskään julkaistu lopullisessa tutkimuksessa, koska perustetta sille ei löytynyt.

3. Automaattisia työkaluja ei saa käyttää muualla kuin NPM:n julkisissa rajapinnoissa pykälän 9 mukaan. Tämä rajoitti tutkimusta esim. kieltämällä verkkosivuihin perustuvan sisältövertailun ja tietojen täydennyksen.

3 Tiedon jäsentäminen ja tutkimusmetodologia

3.1 Yleistä

Toisin kuin keruuvaiheessa, tiedon jäsentämisessä käytettiin sekä Rust-ohjelmointikieltä että Postgres-tietokannan omaa PL/PGSQL-kieltä. Ruby-ohjelmointikieltä kokeiltiin aluksi, mutta se osoittautui raakadatan jäsenysvaiheessa aivan liian hitaaksi – Rust-ohjelmointikielillä saavutettiin jopa kymmenkertainen nopeus jäsennysprosessin aikana. Nopeus mitattiin laskemalla jäsennettyjen pakettikyselyiden määrä suhteessa kuluneeseen aikaan. Lisäksi käsittelyprosessia optimoitiin lisää asentamalla palvelimeen SSD-välimuistilevy, tarjoten tietokannalle mahdollisuuden tallentaa yleisimmin käytetyt tiedot (esim. indeksit) nopeasti luettavaan tallennustilaan.

PL/PGSQL valittiin toiseksi toteutuskieleksi käytännön syistä. Ulkoisen apuohjelman käyttö lisää väistämättä tietyn verran viivettä (esim. TCP/IP-yhteyden viiveet, tyyppimuunnokset Postgres-tyypeistä kielen omiin tyyppeihin), mutta useimmat toimenpiteet eivät vaatineet mitään ulkopuolisia kirjastoja. Lisäksi kieli osoittautui pienen perehtymisen jälkeen hyvin suoraviivaiseksi oppia käyttämään. Näin ollen tietokannan sisällä toteuttaminen osoittautui järkevimmäksi ratkaisuksi – vältetään viiveet tuomalla apuohjelma lähemmäksi dataa, eikä apuohjelmien kirjoittamiseen kulunut ratkaisevasti enemmän aikaa. Kehitysympäristönä Postgres-kyselyille ja -apuohjelmille käytettiin JetBrains'in DataGrip-ohjelmistoa, joka nopeutti kehitysprosessia huomattavasti esim. automaattisella koodintäydennyksellä ja selkeällä graafisella hakukäyttöliittymällä.

3.2 Normalisointi ja kohtuullisen työmäärän rajaaminen

NPM-pakettivaraston metatiedoille (`package.json` jokaisen paketin juurihakemistossa) on määritelty tarkasti JSON-muotoinen rakenne, jota niiden on noudatettava (npm, Inc. 2020d). Kuitenkaan NPM ei ilmeisesti vaadi täydellistä korrektiutta rekisterissä julkaistavilta metatiedoilta, vaan tutkimuksen aikana ilmeni useita erilaisia poikkeavuuksia edellä mainitusta rakenteesta.

Osa näistä poikkeavuuksista on mahdollisesti ajan myötä muuttuneiden käytäntöjen syytä. NPM on ajan myötä kiristänyt nimeämistä koskevia sääntöjä, eikä ole enää mahdollista luoda uusia paketteja jotka ovat liian lähellä (esim. vain kirjainkoko muuttunut) olemassaolevia paketteja (npm, Inc. 2017).

Alkuperäisessä nimiluettelossa oli 1 107 891 pakettia. Nimilista osoittautui kuitenkin epätäydelliseksi, joten alustavan riippuvuuslaskennan jälkeen pakettien metatietoja ladattiin 1 110 726 eri paketista yhteensä. Metatiedoista 1 108 406 pystyttiin rakenteellistamaan. Jäljelle jäävän 2320 paketin metatietoja ei pystytty automaattisesti rakenteellistamaan seuraavista syistä:

- Paketista puuttuu aikatiedot. Tällaiset paketit eivät ole pääosin kiinnostavia, sillä paketeilta puuttuu tällöin tarkka tieto julkaisu- ja muutosajoista. Lisäksi useissa tällaisissa paketeissa ei ole lainkaan julkaistuja versioita, jolloin ne eivät ole kiinnostavia tämän tutkimuksen kannalta. Versioiden määrällä ei muuten ole merkitystä tutkimuksen kannalta.
- Paketin tekstimuotoiset tiedot ovat jollakin tavalla virheellisiä. Yleisin esimerkki tästä on merkkijono joka sisältää nollatavun – jotkut jäsentimet sallivat tämän, mutta tässä tutkimuksessa käytetty Rust-ohjelmointikiellä toteutettu `serde_json` ei.
- Paketin jonkin kentän tietotyyppi on täysin odottamaton. Tästä esimerkkinä kokonaisluku versio-osoittimena, jonka pitäisi yleensä olla merkkijono tai tarkempi kuvaus sopivista versioista.

Näiden pakettien täydellinen käsitteleminen vaatisi runsaasti ylimääräistä työtä. Ottaen huomioon että tällaisten pakettien osuus on noin 0,2%, on tutkimuksen kannalta järkevämpää ajankäyttöä jättää nämä paketit pääosin huomiotta.

Lisäksi paketeissa esiintyi runsaasti erilaisia epäsäännöllisyyksiä, jotka pystyttiin ottamaan huomioon kohtuullisella vaivalla. Seuraava taulukko kuvaa erilaisia havaittuja epäsäännöllisyyssyyppejä ja niiden esiintyvyyttä. Epäsäännöllisyydet eivät sulje toisiaan pois – yhtä tietuetta voi koskea usea epäsäännöllisyyssyyppi.

| Tyyppi | Esiintyvyys | Tarkempi kuvaus |
|---|-------------------------|--|
| Virheellinen merkkijono-henkilökuvaus | 0,29% henkilötietueista | Henkilöä kuvaava merkkijono ei ole säännöllisen rakenteen mukainen. |
| Henkilötietueesta puuttuu nimi | 0,11% henkilötietueista | Henkilötietueessa pitäisi olla aina nimi rakenteen mukaisesti. |
| Tyhjä lisenssitietue | 11% versiotietueista | Lisenssitietue on tyhjä merkkijono. |
| Lisenssittömiä versioita sisältävä paketti | 13% paketeista | Paketti, jossa vähintään yhdestä versiosta puuttuu lisenssi. |
| Nimetön lisenssi | 0,022% versiotietueista | Versiotietueen lisenssillä ei ole oliorakenteen mukaista nimeä. |
| Lisenssi sisältää lisätietoja | 0,19% versiotietueista | Lisenssitieto sisältää tietoja, joita ei ole mainittu pakettimetatietojen rakennekuvauksessa. |
| Epästandardi kotisivu-, kuvaus-, tai lisenssimerkintä | 0,012% versiotietueista | Kotisivu tai kuvaus versiotietueessa eivät ole merkkijonoja, tai lisenssi ei ole merkkijono tai rakennekuvauksen mukainen objekti. |

Taulukko 3: Kuvaus erilaisista pakettimetatiedoissa havaituista epäsäännöllisyyksistä

3.3 Versiot ja versioaikasarjat

3.3.1 Semanttinen versiointi

NPM-pakettivaraston pakettien versiointi noudattaa pääsääntöisesti semanttisen versioinnin käytäntöjä. Semanttinen versiointi määrittelee käytäntöjä siihen, millaiset muutokset vaativat minkäkin tyyppistä versionumeron muutosta (Preston-Werner 2020).

3.3.2 Versioaikasarjojen muodostus ja päivitystahti

Decan, Mens ja Grosjean tutkivat riippuvuusverkostojen evoluutiota seitsemässä eri paketti-järjestelmässä, ja tutkimuksen osana mm. päivitysten selviytymisprosenttia. Toisin sanoen, jos paketti on X kuukautta vanha, kuinka todennäköisesti tähän julkaistaan uusi versio kuukauden kuluessa? (2019).

Tässä tutkimuksessa toteutettiin samankaltainen analyysi käyttämällä PL/PGSQL-kieltä versiodatan aikasarjallistamiseen. Aikasarjassa eroteltiin versiomuutokset semanttisen version sääntöjen mukaan (*major*, *minor*, *patch*), ja lisäksi laskettiin suhteellinen julkaisuaika edelliseen kriteerin täyttävään versioon nähden. Versioiden käyntijärjestyksenä käytettiin versioiden julkaisupäivämääriä, sillä NPM ei pääsääntöisesti pakota mitään tiettyä järjestystä versiotunnisteisiin.

Versioiden julkaisujärjestystä ja versionumeroiden osien erotusta mittayksikköinä on käytetty myös Cox ym. tutkimuksessa. Näistä mittayksiköistä mainitaan useita haittapuolia – versionumeroiden erotusta ei voi helposti tiivistää yhdeksi luvuksi, joka kuvaisi versioiden eroa jollakin järkevällä tavalla. Toisaalta, julkaisujärjestys ei huomioi julkaisutiheyttä, jolloin tiheästi päivittyvät ohjelmistot ovat heikommassa asemassa harvemmin päivittyviin ohjelmitoihin nähden. Version julkaisupäivämäärä puolestaan rankaisee hitaasti päivittyviä ohjelmia, huomioimatta sitä onko tiheille päivityksille todellista tarvetta esim. hyvin vakiintuneissa ja pitkälle kehittyneissä ohjelmissa. (Cox ym. 2015, s.111).

Version julkaisupäivämäärien ja semanttisen versioinnin yhdistäminen voi ehkäistä edellä mainittuja haittapuolia – julkaisupäivämäärät kuvaavat hyvin todenmukaisesti todellista päivittymisnopeutta, mutta toisaalta semanttisella versioinnilla voidaan (semanttista versiointia oikein käyttävissä paketeissa) erottaa päivitysten tyypit toisistaan, jolloin pakettien keskinäinen vertailu on selvästi mielekkäämpää ja helpompaa.

Uudeksi *major*-versioksi laskettiin vain sellainen versio, jonka versionumero oli isoin siihen mennessä havaittu – muissa kategorioissa mikä tahansa muutos riitti. Tarkempi kuvaus algoritmista on liitessä B.

Päivittymistahdin arviointiin puolestaan käytettiin Kaplan-Meier-estimaattoria (Kaplan ja Meier 1958). Kyseinen estimaattori valittiin helpon toteutettavuuden vuoksi, ja tulosten vertailtavuuden vuoksi – Decan, Mens ja Grosjean käyttivät kyseistä estimaattoria omassa tutkimuksessaan (Decan, Mens ja Grosjean 2019, s. 392). Tarkempi kuvaus estimaattorin toteutuksesta PL/PGSQL-kielellä on liitteessä C.

Sarjoja muodostettaessa ilmeni erikoinen poikkeavuus NPM:n aikaleimoissa. Lukuisat päällisin päin eroavat versiot merkittiin samalla aikaleimalla paketin julkaisutiedoissa. Tämä puute koski vain vähemmistöä paketeista, pääosin ennen vuotta 2012 julkaistuja paketteja. Nämä ilmeisen vääristyneet leimaukset suodatettiin pois, jotta tulokset eivät kuvaisi todennukaista nopeampaa päivitystahtia.

3.4 Riippuvuuksien selvittäminen

NPM sallii riippuvuusmääritelmiä, jotka eivät täysin yksiselitteisesti vaadi tiettyä versiota. Näin ollen, on tarpeellista muodostaa aputaulu riippuvuuksien hakua varten – jos tiedämme jonkin tietyn riippuvuusmääritelmän, haluamme saada nopeasti tietoomme sen mitä versiota se vastaisi tietokannassa (npm, Inc. 2020a). On tutkittu, että erityisesti NPM:ssä *caret* (^)-lisämääreen käyttäminen on tyypillistä – *caret* sallii kaikki muut päivitykset paitsi *major*-versionumeroa muuttavat päivitykset (Zerouali ym. 2019, s. 9).

Optimoinnin kannalta on viisasta laskea etukäteen mitä tarkkaa pakettiversiota mikäkin versiomääre vastaa. Näin voi tehdä, koska tutkimus kohdistuu tietyllä ajanhetkellä kerättyyn joukkoon – jos seurattaisiin jatkuvasti NPM:n muutoksia, jouduttaisiin merkittävää osaa määreistä päivittämään vähän väliä. Optimointi rajattiin niihin pakettiversioihin, jotka julkaistiin ennen keräyksen aloittamista.

Versioiden järjestämiseen käytettiin Rustille saatavissa olevaa *semver*-kirjastoa. Mikäli määre ei erikseen määrännyt muuta, selvitettäessä pyrittiin valitsemaan mahdollisimman uusi versio.

3.5 Graafinmuodostus ja -analyysi

Graafianalyysin yhteydessä tutkittiin sitä, miten paketit linkittyvät toisiinsa riippuvuuksien kautta. Seuraavat tutkimuskysymykset olivat erityisen olennaisia graafien osalta:

1. Mitä paketteja käytetään eniten, suorasti tai epäsuorasti?
2. Kuinka pitkiä riippuvuusketjuja kehittyi? Onko havaittavissa jotain lainalaisuuksia, esim. tietyn tyyppisiä riippuvuuksia tai paketteja?
3. Kuinka paljon paketteja eri ”tasoilla” on? Kuinka paljon riippuvuuksien yhteismäärä kasvaa, kun edetään riippuvuusketjussa eteenpäin huomioiden myös transitiiviset riippuvuudet?

Toteutustavaksi valittiin käänteinen tarkastelu, ts. tutkitaan mitkä pakettiversiot riippuvat jostakin tietyistä valitusta versiosta. On helpompaa päätellä mihin paketteihin todennäköisemmin kohdistuu riippuvuuksia, kuin se että mistä paketeista nämä tietyt riippuvuudet lähtevät.

Tähän päätelmään päästiin tutkimalla aiemmin auki laskettuja riippuvuusmääreitä – tiettyihin yleisiin pakettiversioihin kohdistui huomattavia määriä riippuvuuksia, joka osaltaan viittaa jonkinlaisesta klusteroitumisesta. Tämä vastaa myös aiempaa tutkimusta jonkin verran – Decan, Mens ja Grosjeanin mukaan riippuvuudet kohdistuvat pieneen osajoukkoon (Decan, Mens ja Grosjean 2019, s. 398).

Teknisistä ja suorituskyvylisistä rajoitteista johtuen erilaisille versiomääreille tallennettiin vain yksi tarkka, mahdollisimman uusi versio, joten on olennaista valita sopivat lähtöversiot – muussa tapauksessa muodostuva graafi ei välttämättä kuvaisi totuudenmukaisesti paketien suhteita. Kattavaa tilastollista analyysiä ei edellä mainituista syistä myöskään tehty tässä tutkimuksessa, vaan lähinnä kvalitatiivista analyysiä valikoiduista, erityyppisistä lähtötilanteista.

4 Tulokset

4.1 Tilastojen tulkinnasta

On huomioitava, että NPM:n rajapinnan rajoitteiden vuoksi ei ollut mahdollista kaapata koko pakettivaraston tilaa tietyllä yksittäisellä ajanhetkellä. Otos ajoittuu pääosin hieman yli kuukauden mittaiselle ajanjaksolle, joka on saattanut johtaa vääristymiin esim. pakettien versiomäärissä, mikäli paketti on päivittynyt tietojen haun jälkeen.

Mikäli ei erikseen mainita muuta, tiedot perustuvat koko dataan *ilman suodatusta haku-, luonti- tai muokkausaikaleimojen suhteen*.

4.2 Yleiset pakettitilastot

Tässä osiossa esitellään lyhyitä lainalaisuuksia ja johtopäätöksiä, mitä kerätystä datasta voidaan päätellä kokonaisuutena. Paketteihin ja yksittäisiin versioihin pureudutaan tarkemmin myöhemmissä kappaleissa.

4.2.1 Lisenssit

MIT, ISC, ja Apache 2.0-lisenssit nousivat esille suosituimpina lisenseinä. Noin 78,1% kaikista kerätyistä pakettiversioista on lisensoitu jollakin näistä kolmesta lisenssistä, ja tästä osajoukosta noin 53,7% on lisensoitu MIT-lisenssillä. On huomioitava että `npm init`, pakettimetadatan alustuskomento, tarjoaa oletuksena ISC-lisenssiä (npm, Inc. 2020f).

Näissä osuuksissa ei oteta huomioon paketteja, joissa lisenssitieto on joko monikkona (usea mahdollinen lisenssi) tai ilmaistu epäsäännöllisessä muodossa. Tämän vuoksi todellinen osuus voi olla suurempi kuin mitä edellä todettu, sillä esim. GPL esiintyy 27:ssä 100:sta yleisimmistä lisenssimerkinnästä. Yhteensä GPL kuitenkin esiintyy vain 2,8%:ssa pakettiversioista, joten se ei ratkaisevasti vaikuta kolmen ylimmän lisenssin asemaan.

Tarkempi luettelo yleisimmistä lisensseistä on esitetty liitteessä D.

4.2.2 Suosikkimerkinnot

Yksittäinen paketti voidaan suosikkimerkitä, tarkoittaen sitä että joku NPM-käyttäjä lisää sen suosikikseen. Tällainen merkintä ei ole riippuvainen paketin yksittäisistä versioista.

Sata eniten merkittyä pakettia viittaavat siihen, että erityisesti melko yleiskäyttöiset apukirjastot ja -työkalut nauttivat suosiota.

Esimerkkejä järjestyksessä suosituimmista paketeista:

- `express`, apukirjasto HTTP-palvelimen toteuttamiseen
- `request`, apukirjasto HTTP-pyyntöjen tekemiseen
- `lodash`, apukirjasto funktionaaliseen ohjelmointiin
- `gulp`, aputyökalu ohjelmien ja verkkosivujen kääntämiseen ja koontiin

Tarkempi luettelo suosituimmista paketeista on esitetty liitteessä E.

4.2.3 Avainsanat

Ylivoimaisesti suosituin avainsana on `react` 46 020 osumalla yksittäisissä pakettiversioissa (vrt. toinen sija 18 655 osumalla, mielenkiintoista kyllä, sanalla `javascript`), joka viittaa suuren suosion saavuttaneeseen React-käyttöliittymäkirjastoon. Korkeilla sijoilla on myös muita käyttöliittymäkirjastoja – esim. `vue` sijalla 4 ja `angular` sijalla 5.

Erilaiset aputyökalut ja -kirjastot ovat edustettuja suosituimpien avainsanojen joukossa, näistä esimerkkeinä osumamäärien mukaan järjestettynä `typescript`, `webpack`, `yeoman-generator`, `express` ja `eslint`.

Tarkempi luettelo sadasta suosituimmasta avainsanasta on esitetty liitteessä F.

4.2.4 Nimikkeet

Nimikkeistä nousee esille erityisesti TypeScript-tyyppimääritelmien versionimikkeet (esim. `ts3.4`), joille löytyy noin 6000 pakettiosumaa per nimike. Lisäksi nimikkeet `beta` (viittaa beta-versioihin) ja `next` (viittaa nykyistä pääversiota seuraavaan versioon) ovat varsin

suosittuja yli 13 tuhannella osumalla per nimike (npm, Inc. 2020c). Kuitenkin, valtava enemmistö nimikkeistä esiintyy korkeintaan muutamassa paketista koko NPM-pakettivarastossa.

4.2.5 Pakettien versiomäärät

Valtaosalla rekisteristä löytyvistä paketeista ei ole kovin montaa versiota. Noin hieman yli 52.5%:ssa versioita sisältävistä paketeista on korkeintaan 3 versiota, ja 28%:ssa ainoastaan yksi. Vain hieman alle 20%:lla paketeista on yli 10 versiota. Ääripäässä, eniten versioita sisältävällä paketilla oli peräti 7900 julkaistua versiota. Lisäksi 286:lla muulla paketilla oli vähintään 1000 julkaistua versiota.

Versioista noin 239 990 oli julkaistu keräyksen aloittamisen jälkeen. Tämä on hieman yli 2 prosenttia, eli melko pieni määrä. Vain hieman yli neljällä prosentilla kaikista paketeista oli tällaisia versioita.

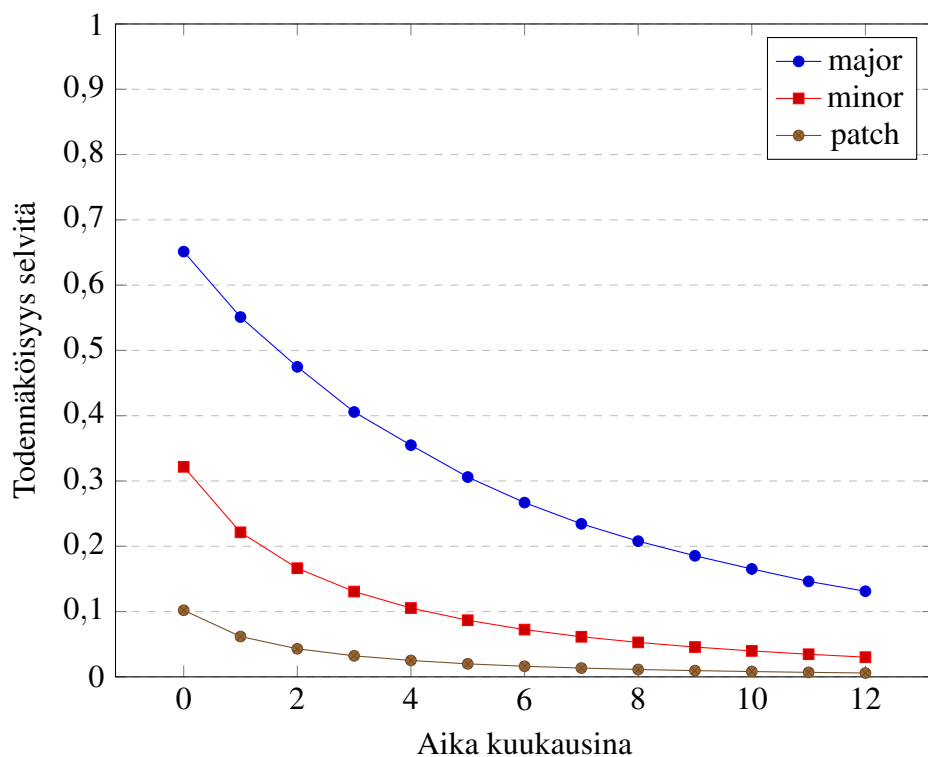
4.3 Pakettiversioiden ominaisuuksia

4.3.1 Versioiden päivittyvyys

Kuviot 2 ja 3 kuvaavat pakettiversioiden arvioitua elinikää. X-akseli esittää aikaa joko päivinä tai kuukausina, ja Y-akseli todennäköisyyttä siihen kestäkö X-akselin antamaa aikaa enemmän, että yksittäinen versio päivittyy. Todennäköisyyden laskemiseen on käytetty Kaplan-Meier-estimaattoria (kts. liite C), ja arvio on laskettu eri semanttisen version tyypeille (*major*, *minor*, *patch*).

Graafeissa on huomioitu ainoastaan ne versiot, jotka lopulta päivittyvät. Mikäli myös päivittymättömät versiot otettaisiin mukaan, olisi todennäköisyys version ”selviämiseksi” myös kauttaaltaan suurempi.

Decan, Mens ja Grosjean totesivat että lähes kaikissa tutkituissa pakettivarastoissa, pakettiversio päivittyy 2 kuukauden sisällä yli 50% todennäköisyydellä. (Decan, Mens ja Grosjean 2019, s. 393). Tämä vastaa tässä havaittuja tuloksia – hitaimmillaankin todennäköisyys alle 2kk päivittyvyydelle on noin 50%. Graafissa on nähtävissä selkeä ero semanttisten versio-tyyppien välillä - pieniä päivityksiä (*minor*, *patch*) tulee selvästi tiheämmin kuin isompia



Kuvio 2: Pakettiversioiden selviytymisennuste 12 kuukauteen asti

(*major*).

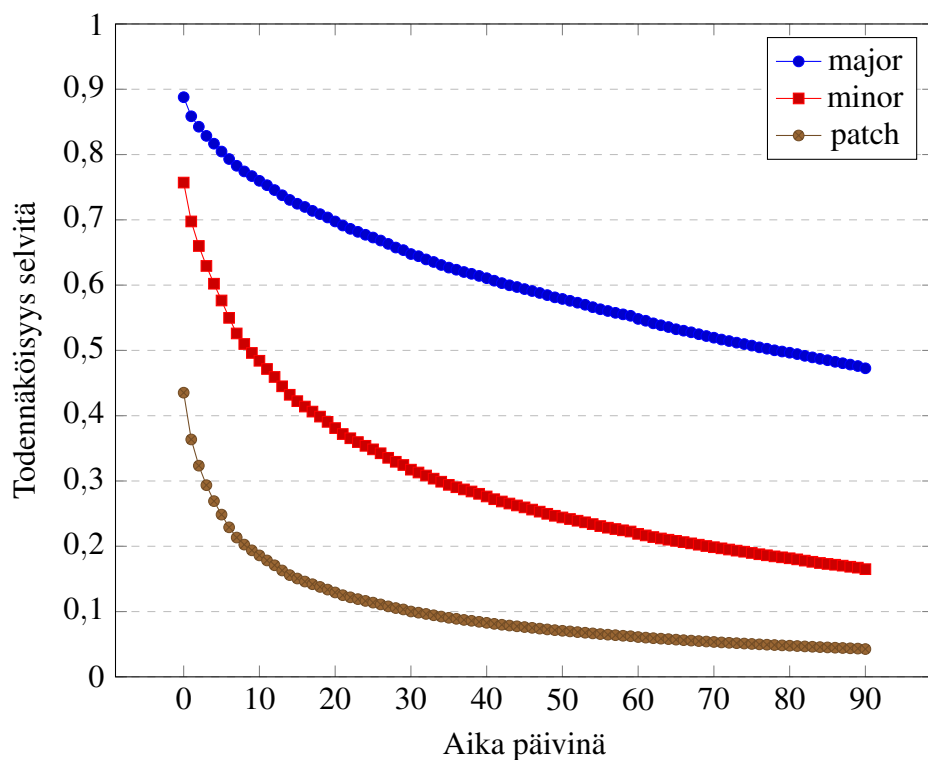
4.3.2 Versiotunnisteet

Koko tietokannassa on yhteensä 586 267 erilaista semanttisen versioinnin mukaista versiotunnistetta. Valtaosa näistä (460 308, noin 79%) on erilaisia esijulkaisuversioita (eng. *pre-release*), jotka sisältävät tekstimuotoisen, piste-erotellun tunnisteen erottaen muuten saman versiotunnisteen omaavat versiot toisistaan.

4.3.3 Riippuvuuksien versiomääreet

Jokaiseen riippuvuuteen liittyy versiomääre, jolla kuvataan se mitä versioita vaativa paketti hyväksyy. Tämä noudattaa aiemmin tekstissä kuvattua semanttisen versioinnin periaatetta. NPM myös hyväksyy pakettivaraston ulkopuolisia riippuvuuksia, esim. Git-repositorioviittauksia.

Kaikilla tietokannasta löytyvillä pakettiversioilla (mukaanlukien myös keräyksen aloituksen

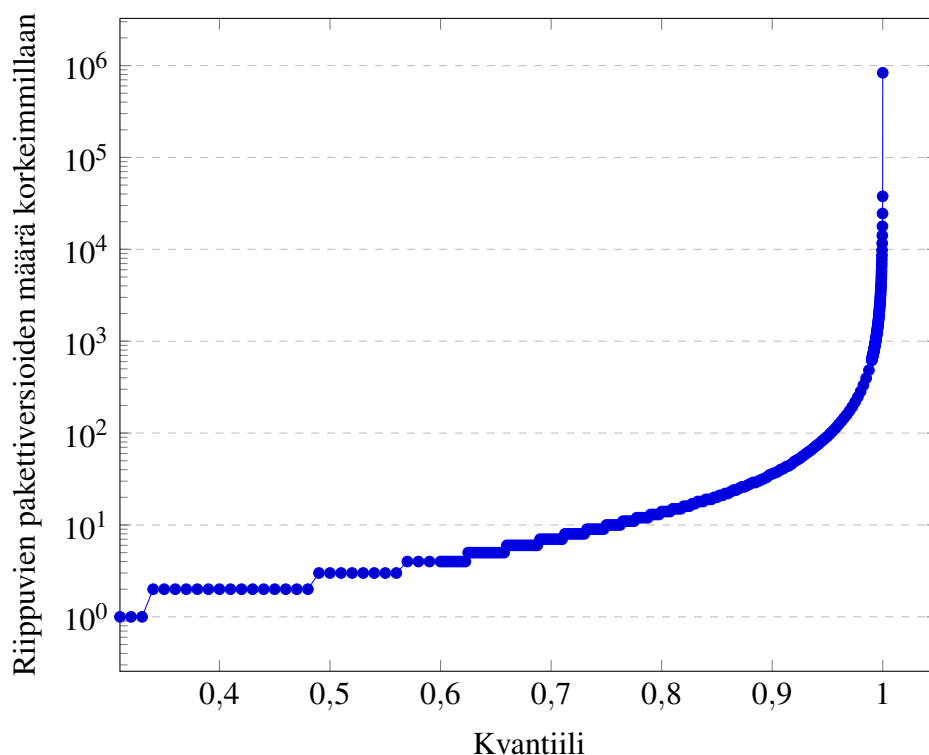


Kuvio 3: Pakettiversioiden selviytymisennuste 90 päivään asti

jälkeen julkaistut) on kaiken kaikkiaan 2 723 226 uniikkia, tietokannasta löytyvään pakettiin osoitettavissa olevaa versiomäärettä. Erityisesti `react`- ja `typescript`-paketteihin kohdistui mitä erilaisimpia versiomääreitä, 3000 ja 2537 erilaista kullekkin. Lisäksi oli 47 191 riippuvuusmäärettä, jotka viittasivat paketteihin joita tietokannasta ei löydy (esim. aiemmin mainitut Git-repositoriot) tai ovat määreinä muuten virheellisiä.

Todennettuissa (ts. pystyttiin määrittämään tarkka versio) riippuvuusmääreissä näkyy selvä jakauma. Pakettiversioiden tavallisista ja valinnaisista riippuvuusmääreistä (poissulkien muut riippuvuustyypit) merkittävä osa vaikuttaisi kohdistuvan vain pieneen osaan kaikista pakettiversioista. Decan, Mens ja Grosjean löysivät osaltaan samankaltaisen tuloksen omassa tutkimuksessaan - riippuvuudet kohdistuvat pääosin pieneen vähemmistöön koko pakettijoukosta (Decan, Mens ja Grosjean 2019, s. 398).

Kuvio 4 havainnollistaa tätä jakaumaa kvantiilien muodossa. Esimerkiksi, noin 33% prosenttiin pakettiversioista kohdistuu vain yksi riippuvuus, ja 99%:iin korkeintaan 618 riippuvuutta.



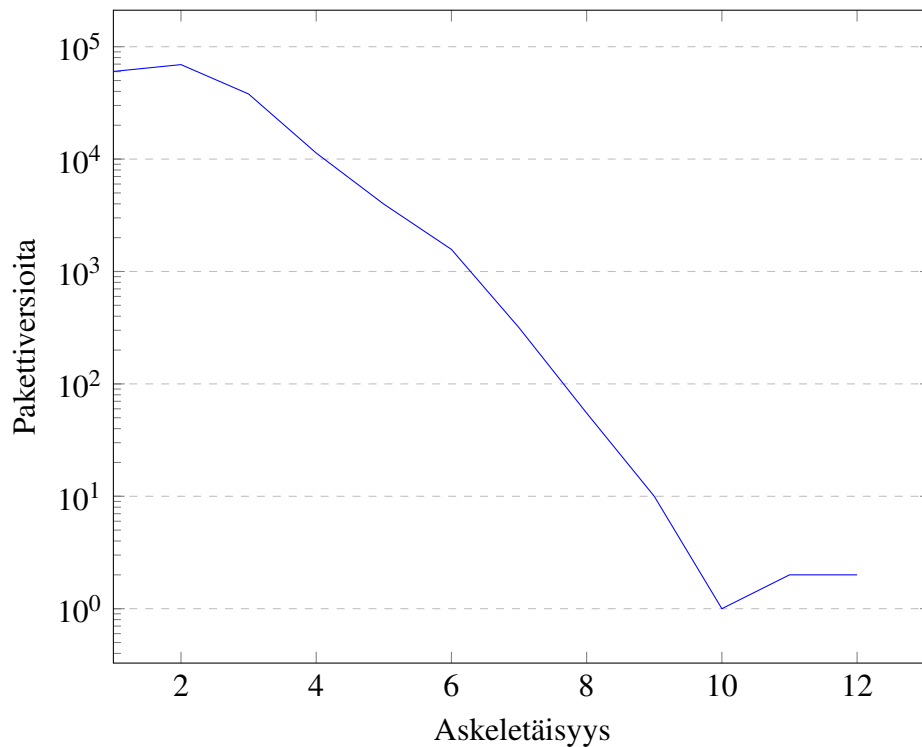
Kuvio 4: Riippuvuusmäärien kvantiilit 0.31-1

4.4 Pakettien riippuvuussuhteet

Kuten aiemmin mainittiin, teknisistä rajoitteista johtuen graafianalyysiin on valittu vain muutamia erityisen kiinnostavia lähtötilanteita. Tarkastelussa tutkittiin vain tavallisia ja valinnaisia riippuvuuksia, siten että lyhyimmäksi poluksi huomioidaan askelmäärältään pienin, kumpaa tahansa riippuvuustyyppiä sisältävä polku. Tietystä paketista pyrittiin valitsemaan vain yksi (yleensä uusin) versio, elleivät riippuvuusmääreet muuhun pakottaneet.

Jokaisesta esimerkistä esitellään kuviona uusien riippuvuuksien määrä askelmäärän mukaan (0. askelella paketit joista analyysi lähti liikkeelle), ja taulukkona askelkohtaiset jakaumat kulloiseltakin askeleelta löytyvistä pakettiversioista muilla askeleilla riippuvien versioiden määrille. Jos esim. taulukossa mainittu minimi jollakin rivillä on 2, se tarkoittaa sitä että jokaisesta kyseisen askeleen pakettiversiosta riippuu vähintään 2 muuta pakettiversiota useimmiten seuraavilla, mutta myös mahdollisesti edellisillä askeleilla.

Mikäli paketin nimi on kirjoitettu koodikirjaisimilla, viitatan nimenomaisesti osiossa käsi-



Kuvio 5: Uusien pakettiversioiden määrä lyhyimmän askeletäisyyden mukaan `lodash@4.17.5`-paketille

teltyyn pakettiin – muussa tapauksessa kirjastoon tai ohjelmaan yleisesti.

4.4.1 `lodash@4.17.5`

Kuten liitteessä G todetaan riippuvuusmäärien mukaan, Lodash on yksi ylivoimaisesti suosituimmista kirjastoista. Lodash-kirjasto on yleiskäyttöinen apukirjasto, joka on suunniteltu erityisesti funktionaalistyyliseen ohjelmointiin (JS Foundation 2020).

Kuten taulukosta 4 voidaan päätellä, käänteiset riippuvuudet selvästi klusteroituvat tiettyihin paketteihin, ja suurimpaan osaan ei tule juurikaan viittauksia.

Huomaamme myös, että ilmeisen suuri osa `lodash`-paketista riippuvista paketeista on lyhyimmillään hyvin lähellä, joko suoraan riippuvana tai yhden väliaskelen kautta. Kuitenkin, ilmeisen merkittävä osa riippuu myös pidemmän matkan päästä. Määrän muutos on suurinta heti 2. askeleen jälkeen, jatkuen karkean logaritmisesti ja pysähtyen lopulta 10. askeleen

| Askel | Minimi | 50% | 75% | 95% | 99% | Maksimi |
|--------------|---------------|------------|------------|------------|------------|----------------|
| 1 | 1 | 0 | 0 | 3 | 15 | 9194 |
| 2 | 0 | 0 | 0 | 2 | 8 | 4582 |
| 3 | 0 | 0 | 0 | 2 | 7 | 2452 |
| 4 | 0 | 0 | 0 | 2 | 9 | 2145 |
| 5 | 0 | 0 | 0 | 3 | 10 | 1028 |
| 6 | 0 | 0 | 0 | 2 | 11 | 80 |
| 7 | 0 | 0 | 0 | 3 | 7 | 22 |
| 8 | 0 | 0 | 0 | 3 | 9 | 9 |
| 9 | 0 | 0 | 1 | 2 | 2 | 2 |
| 10 | 2 | 2 | 2 | 2 | 2 | 2 |
| 11 | 1 | 1 | 1 | 1 | 1 | 1 |
| 12 | 0 | 0 | 0 | 0 | 0 | 0 |

Taulukko 4: lodash@4.17.5-pakettiversion riippuvuusverkon riippuvien pakettien määräjakauma askelittain

kohdalle, josta löytyi ainoastaan yksi paketti.

Tämä on mielenkiintoinen havainto - merkittävä osa `lodash`:in latauksista muodostuu epäsuorista riippuvuuksista, ja iso osa niistä suhteellisen pitkän ketjun kautta. Tämä osaltaan epäilemättä hämärtää loppukäyttäjän ja kehittäjän ymmärrystä siitä, mitä kaikkea yksittäinen tietty ohjelma sisältää ja mistä koodi on peräisin.

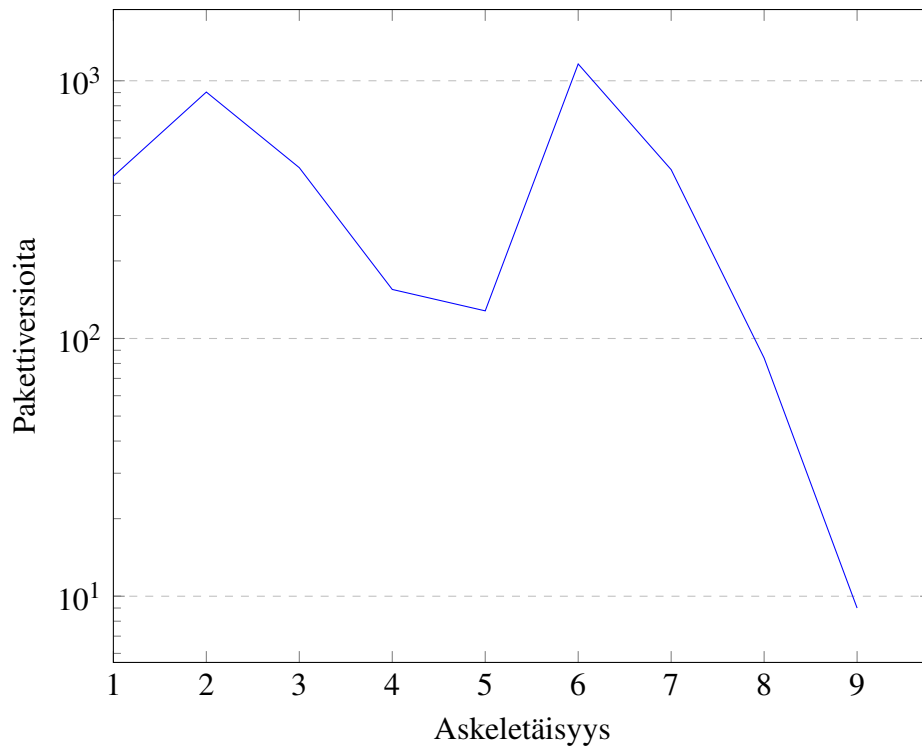
4.4.2 `left-pad@1.3.0`

Tämä paketti valikoitui osittain kuuluisuutensa vuoksi. Paketin kehittäjä poisti pienen, 11 riviä koodia sisältäneen pakettinsa NPM-pakettivarastosta paketin nimeä koskettaneen riidan vuoksi, joka johti lukuisien, muutamien varsin merkittävienkin projektien riippuvuusketjujen rikkoutumiseen (Williams 2020). Tapaus johti mm. NPM:n viralliseen jälkipuintiin ja käytäntöjen tarkentamiseen (npm, Inc. 2016). On myös huomattava ettei `left-pad` ole ainoa esimerkki pienestä riippuvuudesta - esim. `escape-string-regexp`-paketti on 8 riviä pitkä ja koostuu vain yhdestä funktiosta, mutta siitä huhtikuussa 2019 riippui melkein tuhat julkista, tiedossa olevaa pakettia (Cox 2019, s. 2-3).

Ns. *triviaalien* (vähän koodia sisältävien, helposti korvattavien) pakettien käyttöä NPM-ekosysteemissä on tutkittu. Abdalkareemin ym. tutkimuksen mukaan suuri osa kehittäjistä ei näe triviaalien riippuvuuksien käyttöä vahingollisena. Kehittäjät kokevat että julkaistut paketit ovat keskimäärin hyvin testattuja, ja tehostavat ohjelmointityötä. Toisaalta, merkittävä osa kehittäjistä tiedostaa hallitsemattomien riippuvuuskoeloemien ja ohjelmien rikkoutumisen vaaran triviaaleja riippuvuuksia käytettäessä. (2017).

Kuuluisuus ei ollut kuitenkaan ainoa syy - yllämainittuun pakettiversioon kohdistuu 5343 suoraa riippuvuutta, mikä on kvantiiligraafin mukaan varsin huomattava määrä. Lisäksi viikottaisia latauksia on edelleen varsin merkittävästi, pääosin yli 4 miljoonaa kertaa viikossa vuoden 2020 aikana – varsin mielenkiintoista, kun ottaa huomioon että paketti on virallisesti vanhennettu eikä sen käyttöä suositella (npm, Inc. 2020b).

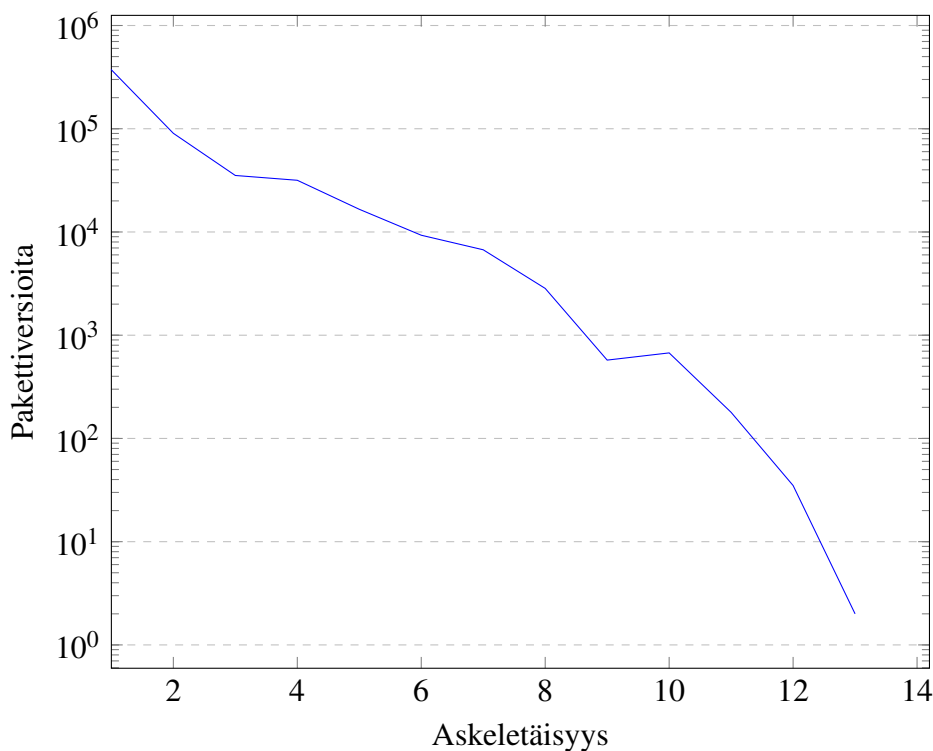
Jakauma näyttää kovin erikoiselta 2. ja 6. askeleen kohdalla. Tarkemmassa tutkimuksessa paljastui syy - pakettiversio `jest@24.9.0` 5. askeleella tuo mukanaan merkittävän määrän riippuvia paketteja yksistään verrattuna muihin paketteihin (1194 riippuvuutta, vrt.



Kuvio 6: Uusien pakettiversioiden määrä lyhyimmän askeletäisyyden mukaan left-pad@1.3.0-paketille

| Askel | Minimi | 50% | 75% | 95% | 99% | Maksimi |
|-------|--------|-----|-----|-----|-----|---------|
| 1 | 0 | 0 | 0 | 2 | 12 | 760 |
| 2 | 0 | 0 | 0 | 2 | 7 | 82 |
| 3 | 0 | 0 | 0 | 2 | 10 | 29 |
| 4 | 0 | 0 | 0 | 2 | 46 | 81 |
| 5 | 0 | 0 | 1 | 3 | 14 | 1194 |
| 6 | 0 | 0 | 0 | 1 | 4 | 283 |
| 7 | 0 | 0 | 0 | 1 | 3 | 39 |
| 8 | 0 | 0 | 0 | 1 | 2 | 2 |
| 9 | 0 | 0 | 0 | 1 | 1 | 1 |

Taulukko 5: left-pad@1.3.0-pakettiversion riippuvuusverkon riippuvien pakettien määräjakauma askelittain



Kuvio 7: Uusien pakettiversioiden määrä lyhyimmän askeletäisyyden mukaan 56% riippuuskvantiilille

`jsdom@11.12.0` 1. askeleelta 760 riippuvalla paketilla ja `left-pad` itse 426 paketilla), ja siitä syystä aiheuttaa piikin.

Tästä voimme päätellä, että yksittäisilläänkin paketeilla voi olla suhteessa huomattava merkitys muiden pakettien käyttömääriin - merkittävä osa `left-pad`:in käyttäjistä saa paketin epäsuorien riippuvuuksien kautta, todennäköisesti suurehkon askelmäärän päästä.

4.4.3 56% kvantiili

Enemmistö riiputuista pakettiversioista on sellaisia, että niihin kohdistuu hyvin vähän riippuvuuksia. Tässä osiossa tarkastellaan kaikkia niitä pakettiversioita joihin kohdistuu alle 3 riippuvuutta, eli karkeasti 56% kvantiilia. Tässä tapauksessa lähdettiin liikkeelle 637 893 eri pakettiversiosta.

| Askel | Minimi | 50% | 75% | 95% | 99% | Maksimi |
|--------------|---------------|------------|------------|------------|------------|----------------|
| 0 | 0 | 1 | 2 | 3 | 3 | 3 |
| 1 | 0 | 0 | 0 | 4 | 13 | 1566 |
| 2 | 0 | 0 | 0 | 13 | 22 | 2558 |
| 3 | 0 | 0 | 0 | 3 | 14 | 24921 |
| 4 | 0 | 0 | 0 | 2 | 13 | 6179 |
| 5 | 0 | 0 | 0 | 3 | 16 | 5856 |
| 6 | 0 | 0 | 0 | 2 | 9 | 5258 |
| 7 | 0 | 0 | 0 | 2 | 11 | 1695 |
| 8 | 0 | 0 | 0 | 2 | 14 | 379 |
| 9 | 0 | 0 | 0 | 4 | 46 | 4135 |
| 10 | 0 | 0 | 0 | 2 | 10 | 712 |
| 11 | 0 | 0 | 0 | 2 | 8 | 80 |
| 12 | 0 | 0 | 0 | 2 | 2 | 2 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 |

Taulukko 6: 56% riippuvuuskvantiilin riippuvuusverkon riippuvien pakettien määräjakauma askelittain

5 Yhteenveto

5.1 Johtopäätöksiä

NPM on pakettivarastona erittäin suuri, ja sen analysointi järjestään hyvin aikaavievää. Kat-
sauksen tekeminen tämänlaiseen suureen ekosysteemiin vaatiikin hyvää teknistä suunnittelua
ja tehokkaiksi, varmatoimiseksi rakennettuja työkaluja. Hieman yllättävää olikin rakenteel-
taan epäsäännöllisten tai tiedoiltaan muulla tavalla poikkeavien (esim. epätarkat julkaisuajat)
pakettien merkittävä määrä. Ilmeisesti validointia ei julkisen NPM-rekisterin puolella tehty
niin tiukasti kuin mitä tutkimuksen alussa oletettiin.

Ekosysteemiin on kehittynyt ilmeisen vahva koodin uudelleenkäytön kulttuuri. Suurin osa
paketeista on lisensoitu vapaan käytön sallivilla lisensseillä, ja riippuvuustarkastelun mu-
kaan on varsin tavanomaista käyttää ulkoisia riippuvuuksia – riippuvuuksien kohteet jakau-
tavat tosin varsin terävästi suosittuihin paketteihin. Uudelleenkäytön käänköpuolena on riip-
puvuusketjujen kasvaminen hankalasti hallittavan kokoiseksi, ja niin pitkiksi ettei kehittäjän
ole välttämättä helppo hahmottaa mistä osista hänen ohjelmansa lopulta koostuu. Tämä voi
olla turvallisuusriski, kuten Muranen kuvaa – useita suosittuja paketteja on erinäisin konstein
tartutettu haittaohjelmilla, jolloin hyökkääjällä on ollut mahdollisuus suorittaa omaa koodi-
aan käyttäjän ympäristössä (Muranen 2019, s. 18).

Suurimmasta osasta paketteja ei ole julkaistu kuin muutama versio, mutta toisaalta kehitty-
vät paketit keskimäärin päivittyvät varsin virkeästi. Kuten riippuvuuksissakin, myös päivit-
tyvyydessä on havaittavissa selvää jakautumista.

5.2 Jatkotutkimusideoita

Pelkästään tarpeellisen tutkimusvälineistön kehittämiseen meni melko runsaasti aikaa, joka
osaltaan rajoitti gradun puitteissa tehtävissä olevaa tutkimusta. Toisaalta, tutkimuksen yhtey-
dessä selvisi myös useita tutkimusaiheita, jotka voisivat palvella hyvin laajennoksina tämän
tutkimuksen aihepiirille.

5.2.1 Visualisointi

Riippuvuuksien visualisointi epäilemättä helpottaisi ketjujen hahmottamista, mutta riippuvuusketjujen keskimääräinen suuri koko tekee kuvien muodostamisesta haastavaa. Tutkimuksen yhteydessä kehitettiin ohjelma DOT-kielisten kuvaustiedostojen muodostamiseen, mutta GraphViz:in rajoitteet tulivat vastaan (suhteessa) melko pienilläkin riippuvuusgraafeilla.

Valtaviin, yksityiskohtarikkaiden graafien muodostamiseen erikoistuneelle työkalulle voisi olla kysyntää NPM-ekosysteemin sisällä – tähän voisi mahdollisesti soveltaa esim. *International Symposium on Graph Drawing and Network Visualization*-konferenssissa esiteltyjä tutkimustuloksia. Tämä olisi erityisen hyödyllistä yhdistettynä tehokkaaseen graafien päivitysmenetelmään, jotta NPM-pakettivaraston muutoksia voitaisiin jatkuvasti seurata. NPM tarjoaa valmiiksi rajapinnan jolla muutoksia voidaan seurata reaaliaikaisesti, joten seuranta on mahdollista toteuttaa tehokkaasti (npm, Inc. 2020e).

5.2.2 Metatiedot ja koodi

Tämä tutkimus keskittyi lähinnä pakettien metatietoihin, ja pakettien koodi jätettiin kokonaan huomiotta. Myös erilaiset NPM-pakettivaraston ulkopuoliset viittaukset (esimerkiksi riippuvuusviittaukset Git-versionhallintajärjestelmiin) sivuutettiin täysin. Erilaisten pakettiosajoukkojen koodiin perustuva tutkimus voisi paljastaa lisää mielenkiintoisia yksityiskohtia esim. riippuvuuksien toiminnallisen merkittävyyden (paljonko paketti tuo ns. "uutta" koodia) ja pakettien sisäisen rakenteen (esim. kuinka nykyaikaisia JS-rakenteita koodi käyttää) suhteen.

Muranen totesi omassa tutkimuksessaan turvallisen pakettienhallinnan työkalujen olevan kehittymättömiä (Muranen 2019, s. 60), joten ainakin turvallisuusanalyysiin käytettäville työkaluille on melko varmasti kysyntää. Tämä kuitenkin vaatii huomattavasti enemmän resursseja kuin mitä yksittäisen gradun aikana on käytettävissä, pelkästään jo tallennustilan ja prosessointikapasiteetin kannalta, puhumattakaan analyysityökalujen kehittämiseen vaadittavasta työpanoksesta.

Lähteet

Abdalkareem, Rabe, Olivier Nourry, Sultan Wehaibi, Suhaib Mujahid ja Emad Shihab. 2017. “Why Do Developers Use Trivial Packages? An Empirical Case Study on Npm”. Teoksessa *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*, 385–395. ESEC/FSE 2017. Paderborn, Germany: Association for Computing Machinery. ISBN: 9781450351058. doi:10.1145/3106237.3106267. <https://doi.org/10.1145/3106237.3106267>.

Amery, Mark. 2018. “node.js – List all public packages in the NPM registry – Stack Overflow”. Tammikuu. Viitattu 28. huhtikuuta 2020. <https://stackoverflow.com/questions/48251633/list-all-public-packages-in-the-npm-registry>.

Apache Foundation. 2020. “1.3.2 /db/_all_docs - Apache CouchDB 3.0 Documentation”. Viitattu 28. huhtikuuta 2020. <https://docs.couchdb.org/en/stable/api/database/bulk-api.html>.

Carbonnelle, Pierre. 2020. “PYPL PopularitY of Programming Language index”. Viitattu 22. huhtikuuta 2020. <http://pypl.github.io/PYPL.html>.

Cox, J., E. Bouwers, M. v. Eekelen ja J. Visser. 2015. “Measuring Dependency Freshness in Software Systems”. Teoksessa *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, 2:109–118. Toukokuu. doi:10.1109/ICSE.2015.140.

Cox, Russ. 2019. “Surviving Software Dependencies”. *Commun. ACM* (New York, NY, USA) 62, numero 9 (elokuu): 36–43. ISSN: 0001-0782. doi:10.1145/3347446. <https://doi.org/10.1145/3347446>.

Decan, Alexandre, Tom Mens ja Philippe Grosjean. 2019. “An empirical comparison of dependency network evolution in seven software packaging ecosystems”. *Empirical Software Engineering* 24, numero 1 (helmikuu): 381–416. ISSN: 1573-7616. doi:10.1007/s10664-017-9589-y. <https://doi.org/10.1007/s10664-017-9589-y>.

Hyvä tieteellinen käytäntö. 2012. Tutkimuseettinen neuvottelukunta.

JS Foundation. 2020. Viitattu 27. huhtikuuta 2020. <https://lodash.com/>.

Kaplan, E. L., ja Paul Meier. 1958. “Nonparametric Estimation from Incomplete Observations”. *Journal of the American Statistical Association* 53 (282): 457–481. doi:10.1080/01621459.1958.10501452. eprint: <https://www.tandfonline.com/doi/pdf/10.1080/01621459.1958.10501452>. <https://www.tandfonline.com/doi/abs/10.1080/01621459.1958.10501452>.

Muranen, Markus. 2019. “Security principles for package management as part of software development lifecycle and processes”, Jyväskylän yliopisto. <https://jyx.jyu.fi/handle/123456789/66197>.

npm, Inc. 2016. “kik, left-pad, and npm”. Maaliskuu. Viitattu 28. huhtikuuta 2020. <https://blog.npmjs.org/post/141577284765/kik-left-pad-and-npm>.

———. 2017. “New Package Moniker rules”. Joulukuu. Viitattu 28. huhtikuuta 2020. <https://blog.npmjs.org/post/168978377570/new-package-moniker-rules>.

———. 2019a. “npm Open-Source Terms”. Elokuu. Viitattu 28. huhtikuuta 2020. <https://www.npmjs.com/policies/open-source-terms>.

———. 2019b. “the npm registry documentation repo”. Elokuu. Viitattu 28. huhtikuuta 2020. <https://github.com/npm/registry/tree/master/docs>.

———. 2020a. “About semantic versioning | npm Documentation”. Viitattu 28. huhtikuuta 2020. <https://docs.npmjs.com/about-semantic-versioning>.

———. 2020b. “left-pad - npm”. Viitattu 22. huhtikuuta 2020. <https://www.npmjs.com/package/left-pad>.

———. 2020c. “npm dist-tag | npm Documentation”. Viitattu 3. toukokuuta 2020. <https://docs.npmjs.com/cli/dist-tag>.

———. 2020d. “npm-package.json – Specifics of npm’s package handling”. Tammikuu. Viitattu 28. huhtikuuta 2020. <https://docs.npmjs.com/files/package.json>.

———. 2020e. “so you want to write a follower”. Viitattu 4. toukokuuta 2020. <https://github.com/npm/registry/blob/master/docs/follower.md>.

npm, Inc. 2020f. “using-npm | config | More than you probably want to know about npm configuration”. Viitattu 3. toukokuuta. <https://github.com/npm/cli/blob/latest/docs/content/using-npm/config.md>.

“Open Data - Libraries.io”. 2020. Viitattu 28. huhtikuuta. <https://libraries.io/data>.

Preston-Werner, Tom. 2020. “Semantic Versioning 2.0.0”. Viitattu 28. huhtikuuta 2020. <https://semver.org/spec/v2.0.0.html>.

Williams, Chris. 2020. “How one developer just broke Node, Babel and thousands of projects in 11 lines of JavaScript”. Viitattu 22. huhtikuuta. https://www.theregister.co.uk/2016/03/23/npm_left_pad_chaos.

Zerouali, Ahmed, Tom Mens, Jesus Gonzalez-Barahona, Alexandre Decan, Eleni Constantinou ja Gregorio Robles. 2019. “A formal framework for measuring technical lag in component repositories — and its application to npm”. E2157 smr.2157, *Journal of Software: Evolution and Process* 31 (8): e2157. doi:10.1002/smr.2157. <https://onlinelibrary.wiley.com/doi/abs/10.1002/smr.2157>.

Liitteet

A SQL-tietokantarakenteen skeema pääpiirteittäin

```
1 -- Tästä tiedostossa on luetteloitu vain keskeisimmät taulurakenteet.
  ↳ Väliaikaistaulut, näkymät (engl. "view"), useimmat indeksit,
  ↳ proseduurit ja tietokannan tekniset oikeustiedot on jätetty pois
  ↳ selkeyden vuoksi.

2
3 -- Raaka kyselydata, yksi rivi vastaa yhtä hakukertaa.
4 create table raw_query_data
5 (
6     id bigint not null
7     pkg_name text, -- Paketin nimi
8     queried_on timestamp with time zone, -- Milloin haettu
  ↳ NPM-rekisteristä
9     query_json text, -- Rekisterin vastaus sellaisenaan
10    meta_json text -- Verkkolevyiltä sellaisenaan tuodut metatiedot
  ↳ mahdollista myöhempää tarkastelua varten
11 );
12
13 -- Rakenteellistettu pakettidata, yksi rivi vastaa yhtä pakettia
14 create table package
15 (
16     id bigserial not null
17         constraint package_pkey
18         primary key,
19     name text not null, -- Paketin nimi
20     npm_create_date timestamp with time zone, -- Rekisterin mukainen
  ↳ luontipäivämäärä
21     npm_modification_date timestamp with time zone, -- Rekisterin
  ↳ mukainen muokauspäivämäärä
22     fetchnpm_query_date timestamp with time zone, --
  ↳ fetch-npm-list-hakuohjelman hakupäivämäärä (vastaa
  ↳ RAW_QUERY_DATA-taulun "queried_on"-saraketta)
23     based_on_rq bigint -- Viittaus tätä pakettia vastaavaan
  ↳ raakadatariviin, jos olemassa
```



```

24         constraint package_based_on_rq_fkey
25             references raw_query_data
26             on update cascade on delete cascade
27     );
28
29     -- Vaadi että paketin nimi on aina uniikki
30     create unique index package_uniq_name
31         on package (name);
32
33     -- Pakettiversiot - yksi rivi vastaa yhtä
34     ↪ "package_reference"-sarakeessa viitatus paketin versiota
35     create table package_version
36     (
37         id bigserial not null
38             constraint package_version_pkey
39                 primary key,
40         version_name text not null, -- Semanttinen versiotunniste, esim.
41         ↪ "1.1.10.patch-1"
42         package_reference bigint not null -- Viittaus tämän version
43         ↪ sisältävään pakettiin
44             constraint package_version_package_reference_fkey
45                 references package
46                 on update cascade on delete cascade,
47         description text not null, -- Paketin vapaamuotoinen kuvaus
48         license text not null, -- Paketin SPDX-lisenssi
49         author bigint -- Viittaus paketin tekijään
50             constraint package_version_users_id_fk
51                 references users,
52         deprecated text, -- Paketin vanhentumistieto, jos tiedossa
53         homepage text, -- Kotisivu, jos tiedossa
54         bug_reports text, -- Bugiraporttien osoite, jos tiedossa
55         repository text, -- Lähdekoodin repositorio, jos olemassa
56         version_date timestamp with time zone not null, -- Version
57         ↪ julkaisupäivämäärä
58         over_date boolean default false not null -- Onko pakettiversio
59         ↪ julkaistu keräyksen aloituspäivänä tai sen jälkeen
60     );

```

```

56
57 -- Pakettiversioiden OVER_DATE-sarake luotiin seuraavasti
58 BEGIN TRANSACTION;
59     ALTER TABLE PACKAGE_VERSION ADD COLUMN over_date BOOLEAN NOT
        → NULL DEFAULT false;
60     UPDATE package_version SET over_date=true WHERE
        → version_date>=to_timestamp('20-10-2019', 'DD-MM-YYYY');
61     CREATE INDEX package_version_over_date ON
        → PACKAGE_VERSION(over_date);
62 COMMIT;
63
64 -- Pakettien nimikkeet. Yksi rivi vastaa yhtä nimikettä määrätyssä
        → versiossa.
65 create table package_dist_tag
66 (
67     tag_name text not null, -- Nimike tekstimuotoisena
68     package_version bigserial not null -- Viittaus tätä koskevaan
        → pakettiversioon.
69         constraint package_dist_tag_package_version_fkey
70             references package_version
71             on update cascade on delete cascade
72 );
73
74 -- Pakettien ulkoiset riippuvuudet (ei tietokannassa yksilöitävänä
        → pakettina). Yksi rivi on yksi riippuvuusmerkintä.
75 create table package_external_dependency
76 (
77     dependency_kind text not null, -- Riippuvuuden tyyppi
78     depended_towards text not null, -- Riipputtavan paketin nimi
79     depended_by bigserial not null -- Riippuvan paketin tunniste
80         constraint package_external_dependency_depended_by_fkey
81             references package_version
82             on update cascade on delete cascade,
83     version_spec text not null -- Versiomääre
84 );
85

```

```

86 -- Pakettien tunnetut riippuvuudet (voidaan osoittaa pakettiviittaus
    → tietokannan sisällä). Yksi rivi on yksi riippuvuusmerkintä.
87 create table package_known_dependency
88 (
89     dependency_kind text not null, -- Riippuvuuden tyyppi
90     depended_by bigserial not null -- Riippuvan paketin tunniste
91         constraint package_known_dependency_depended_by_fkey
92         references package_version
93         on update cascade on delete cascade,
94     version_spec text not null, -- Versiomääre
95     depends_on bigserial not null -- Riipputtavan paketin tunniste
96         constraint package_known_dependency_depends_on_fkey
97         references package
98         on update cascade on delete cascade
99 );
100
101 -- Pakettien suosikkimerkinnät - yksi rivi on yksi merkintä
102 create table package_favs
103 (
104     npm_username text not null, -- Käyttäjänimi
105     package bigserial not null -- Viittaus pakettiin
106         constraint package_favs_package_fkey
107         references package
108         on update cascade on delete cascade
109 );
110
111 -- Paketin avainsanat - yksi rivi on yksi avainsana
112 create table package_keyword
113 (
114     keyword text not null, -- Avainsana
115     package_reference bigserial not null -- Viittaus pakettiin
116         constraint package_keyword_package_reference_fkey
117         references package
118         on update cascade on delete cascade
119 );
120
121 -- NPM-käyttäjät - yksittäinen rivi vastaa yhtä käyttäjää

```

```

122 create table users
123 (
124     name text,
125     id bigserial not null
126         constraint users_pkey
127             primary key,
128     email text,
129     url text
130 );
131
132 -- Käyttäjä-versio-relaatio. Yksi rivi vastaa yhtä suhdetta käyttäjän ja
133   ↪ pakettiversion välillä (esim. ylläpitäjä, osallistuja)
134 create table user_version_relation
135 (
136     "user" bigserial not null -- Viittaus käyttäjään
137         constraint user_version_relation_user_fkey
138             references users
139                 on update cascade on delete cascade,
140     package_version bigserial not null -- Viittaus pakettiversion
141         constraint user_version_relation_package_version_fkey
142             references package_version
143                 on update cascade on delete cascade,
144     relation_kind text not null -- Suhteen tyyppi
145 );
146
147 -- Aukipuretut versiot. Semanttisia versiotunnisteita on hankala
148   ↪ käsitellä suoraan SQL-kielellä, joten esim. versioiden vertailua
149   ↪ varten luotiin erityinen taulu.
150 create table versions_unpacked
151 (
152     version_name text not null -- Versiotunniste kokonaisuudessaan
153     ↪ merkkijonona
154         constraint versions_unpacked_pkey
155             primary key,
156     major bigint not null, -- Semanttisen version tunnisteet
157     minor bigint not null,
158     patch bigint not null,

```

```

155     pre text, -- Esiversion tunniste, jos versiotunnisteessa
156     build text -- Koontiversio, jos tunnisteessa
157 );
158
159 -- Versioaikasarjataulu. Yksi rivi vastaa yhtä versioaikasarjan osaa, ja
    ↪ rivin perusavain määräytyy versioviittauksen, järjestysluvun ja
    ↪ tapahtuman mukaisesti
160 create table version_time_sequence
161 (
162     package bigint not null -- Viittaus pakettiin
163         constraint version_time_sequence_package_fkey2
164         references package,
165     version_reference bigint not null -- Viittaus pakettiversioon
166         constraint version_time_sequence_version_reference_fkey2
167         references package_version,
168     date timestamp with time zone not null, -- Tapahtuman päivämäärä
169     flag text not null, -- Tapahtuma (esim. uusi major-versio, uusi
    ↪ minor-versio)
170     order_sequence bigint not null, -- Järjestysluku, vanhin
    ↪ tapahtuma on pienimmällä luvulla
171     offset_from_creation interval not null, -- Kulunut aika paketin
    ↪ luomisesta
172     offset_from_major_version interval, -- Kulunut aika
    ↪ viimeisimmästä edeltävästä (major-versionumeroltaan
    ↪ pienemmästä tai samasta) major-versiosta
173     offset_from_minor_version interval, -- Kulunut aika
    ↪ viimeisimmästä edeltävästä minor-versiosta
174     offset_from_patch_version interval, -- Kulunut aika
    ↪ viimeisimmästä edeltävästä patch-versiosta
175     offset_from_any_version interval, -- Kulunut aika mistä tahansa
    ↪ edeltävästä versiosta
176
177     constraint version_time_sequence_pkey2 -- Taulu käyttää
    ↪ komposiittiperusavainta
178         primary key (version_reference, order_sequence, flag)
179 );
180

```

```

181 -- Virheelliset riippuvuusmääreet. Yksi rivi vastaa yhtä löydettyä
    ↳ virheellistä riippuvuusmäärettä
182 create table bad_dep_specifiers
183 (
184     dep_on_name text not null, -- Viitatus pakettin nimi
185     dep_on_pkg_id bigint -- Viitatus pakettin tunniste, jos tiedossa
186         constraint bad_dep_specifiers_dep_on_pkg_id_fkey
187         references package,
188     version_specifier text not null, -- Versiomääre
189     explanation text not null -- Kuvaus siitä miksi määre on
    ↳ virheellinen
190 );
191
192 -- Riippuvuusmääreiden konkreettiset kohdeversiot. Yksi rivi vastaa
    ↳ yhden riippuvuusmääreen tulkintaa tietylle paketille
193 create table dependency_pair_target_versions
194 (
195     dep_on bigint not null -- Viittaus riiputtuun pakettiin
196         constraint dependency_pair_target_versions_dep_on_fkey
197         references package,
198     version_spec text not null, -- Versiomääre
199     target_version bigint -- Konkreettinen kohdeversio, jos
    ↳ mahdollista määrittää
200         constraint
    ↳ dependency_pair_target_versions_target_version_fkey
201         references package_version,
202     constraint dependency_pair_target_versions_pkey
203         primary key (dep_on, version_spec)
204 );

```

B Versioaikasarjojen luontialgoritmi PL/PGSQL-kielillä

```
1  -- Tuo kaikki paketit, joilla on keräyspäivämäärän alittavia versioita
2  create function vts_ingest_all() returns void
3      language plpgsql
4  as $$
5  DECLARE
6      data_query record;
7  BEGIN
8      FOR data_query IN
9          SELECT P.npm_create_date, P.id FROM PACKAGE P WHERE
            ↳ EXISTS (SELECT * FROM PACKAGE_VERSION PV WHERE
            ↳ PV.package_reference=P.id AND PV.over_date=false)
            ↳ AND NOT EXISTS (SELECT * FROM VERSION_TIME_SEQUENCE
            ↳ VTS WHERE VTS.package=P.id)
10     LOOP
11         PERFORM VTS_INGEST_PACKAGE(data_query.id,
            ↳ data_query.npm_create_date);
12     END LOOP;
13 END;
14 $$;
15
16 -- Tuo yksittäinen paketti versioaikasarjoihin
17 create function vts_ingest_package(pid bigint, create_date timestamp
            ↳ with time zone) returns void
18     language plpgsql
19 as $$
20 DECLARE
21     -- Nykyinen järjestysluku
22     cur_order_seq BIGINT := 0;
23     -- Mikä on viimeisimmän huomaamame, määrättyä tyyppiä
            ↳ (major, minor, patch, mikä tahansa) olevan version
            ↳ julkaisupäivämäärä
24     major_version_date TIMESTAMP WITH TIME ZONE := null;
25     minor_version_date  TIMESTAMP WITH TIME ZONE := null;
26     patch_version_date  TIMESTAMP WITH TIME ZONE := null;
27     last_version_date   TIMESTAMP WITH TIME ZONE := null;
28
```

```

29      --- Viimeisimmät löydetyt versiot osittain
30      last_major_version BIGINT := null;
31      last_minor_version BIGINT := null;
32      last_patch_version BIGINT := null;
33      -- Olemmeko jo saavuttaneet version 1.0.0
34      first_major_version_reached boolean := false;
35
36      -- Välimuuttuja versiokyselylle
37      versions_query record;
38
39      BEGIN
40      -- Valitse versiot julkaisujärjestyksessä, olettaen että
41      --   ↳ VERSIONS_UNPACKED-taulusta löytyy tulkinta kaikille
42      --   ↳ versiotunnisteille
43      FOR versions_query IN
44      SELECT PV.id, PV.version_name, PV.version_date,
45      --   ↳ VP.major, VP.minor, VP.patch, VP.pre FROM
46      --   ↳ PACKAGE_VERSION PV
47      LEFT JOIN VERSIONS_UNPACKED VP ON
48      --   ↳ VP.version_name=PV.version_name
49      WHERE PV.package_reference=pid AND PV.over_date=false
50      ORDER BY PV.version_date ASC
51
52      LOOP
53      DECLARE
54      major_version_changed boolean := false;
55      minor_version_changed boolean := false;
56      patch_version_changed boolean := false;
57      indicate_first_major_version boolean := false;
58
59      BEGIN
60      -- Merkitse major-versio vain, jos se on korkeampi
61      --   ↳ kuin edellinen versio (jos edellistä versiota
62      --   ↳ on), ja major-luku on isompi kuin 0
63      IF (versions_query.major >= 1 AND
64      --   ↳ (last_major_version IS NULL OR
65      --   ↳ versions_query.major > last_major_version)) THEN
66      major_version_changed := true;
67      -- Onko saavutettu versio 1.0.0?
68      IF (first_major_version_reached = false) THEN

```



```

57         indicate_first_major_version := true;
58     END IF;
59
60     -- Nollaa pienemmät versiot, jos pääversio meni
61     ↪ taaksepäin.
62     -- (HUOM: Ylemmän ehdon vuoksi ei mahdollista,
63     ↪ aiemmin ylempi ehto ei pakottanut
64     ↪ versionumeron kasvua)
65     --IF (last_major_version IS NOT NULL AND
66     ↪ versions_query.major < last_major_version)
67     ↪ THEN
68     --     minor_version_date = NULL;
69     --     patch_version_date = NULL;
70     --end if;
71
72     END IF;
73
74     -- Nämä käyttäytyvät samoin miten "major_version"
75     ↪ aiemmin - mikäli versio menee taaksepäin,
76     ↪ kohtelee nollautuneena.
77
78     IF (versions_query.minor != last_minor_version AND
79     ↪ last_minor_version IS NOT NULL) OR
80     ↪ (last_minor_version IS NULL and
81     ↪ versions_query.minor != 0) OR
82     ↪ major_version_changed THEN
83         minor_version_changed := true;
84         IF (last_minor_version IS NOT NULL AND
85         ↪ versions_query.minor < last_minor_version)
86         ↪ THEN
87             minor_version_date = NULL;
88             patch_version_date = NULL;
89         end if;
90     END IF;

```

```

77      IF (versions_query.patch != last_patch_version AND
      ↪ last_patch_version IS NOT NULL) OR
      ↪ (last_patch_version IS NULL AND
      ↪ versions_query.patch != 0) OR
      ↪ minor_version_changed THEN
78          patch_version_changed := true;
79      IF (last_patch_version IS NOT NULL AND
      ↪ versions_query.patch < last_patch_version)
      ↪ THEN
80          patch_version_date = NULL;
81      end if;
82  END IF;
83
84  DECLARE
85      indication_array text[] :=
      ↪ ARRAY['version_changed']; -- Mitä tapahtumia
      ↪ haluamme ilmoittaa
86      indication text; -- Silmukan muuttuja
87  BEGIN
88      -- Muodosta indikaatiot saatujen tietojen mukaan
89      IF indicate_first_major_version THEN
90          indication_array :=
          ↪ array_append(indication_array,
          ↪ 'first_major_version');
          first_major_version_reached := true;
91      END IF;
92
93
94      IF major_version_changed THEN
95          indication_array :=
          ↪ array_append(indication_array,
          ↪ 'major_version');
96          IF versions_query.major >= 1 THEN
97              indication_array :=
              ↪ array_append(indication_array,
              ↪ 'major_version_over_1');
98          end if;
99      END IF;

```

```

100     IF minor_version_changed THEN
101         indication_array :=
            ↳ array_append(indication_array,
            ↳ 'minor_version');
102     END IF;
103     IF patch_version_changed THEN
104         indication_array :=
            ↳ array_append(indication_array,
            ↳ 'patch_version');
105     END IF;
106
107     -- Lisää tapahtumat tietokantaan
108     -- Muuttuva versio sisältyy laskettuun aikaan -
            ↳ jos esim. major-versio muuttunut, merkitty
            ↳ aika on nolla.
109     FOREACH indication IN ARRAY indication_array
110     LOOP
111         INSERT INTO VERSION_TIME_SEQUENCE VALUES (
112             pid,
113             versions_query.id,
114             versions_query.version_date,
115             indication,
116             cur_order_seq,
117             age(versions_query.version_date, create_d|
            ↳ ate),
118             age(versions_query.version_date,
            ↳ major_version_date),
119             age(versions_query.version_date,
            ↳ minor_version_date),
120             age(versions_query.version_date,
            ↳ patch_version_date),
121             age(versions_query.version_date,
            ↳ last_version_date)
122         );
123     END LOOP;
124
125     -- Muuta päivämäärät ja versiot jos tarpeen

```

```

126         IF major_version_changed THEN
127             last_major_version := versions_query.major;
128             major_version_date :=
                ↪ versions_query.version_date;
129         END IF;
130         IF minor_version_changed THEN
131             last_minor_version := versions_query.minor;
132             minor_version_date :=
                ↪ versions_query.version_date;
133         END IF;
134         IF patch_version_changed THEN
135             last_patch_version := versions_query.patch;
136             patch_version_date :=
                ↪ versions_query.version_date;
137         END IF;
138
139
140         last_version_date := versions_query.version_date;
141
142         -- Kasvata järjestyksukua
143         cur_order_seq := cur_order_seq + 1;
144     END;
145 END;
146 END LOOP;
147 END;
148 $$;

```

C Esimerkki Kaplan-Meier-estimaattorista PL/PGSQL-kielillä

```
1  -- Laskee Kaplan-Meier-estimaattorilla eloonjäämisarvion
   ↳ "major"-versionumeroiden mukaan. Tämä versio laskee ajan kuukausina
2  create function calculate_kaplan_major()
3      returns TABLE(month bigint, probability double precision)
4      language plpgsql
5  as
6  $$
7  DECLARE
8      probability double precision := 1.0;
9      ti bigint := 0;
10     ni_total_versions bigint;
11  BEGIN
12
13     -- Hae soveltuvat versioaikasarjojen tiedot
14     WITH next_steps AS (
15         SELECT vts.package, vts.order_sequence
16         FROM version_time_sequence vts WHERE
17             vts.offset_from_major_version IS NOT NULL
18             AND vts.offset_from_major_version != '0 seconds' --
           ↳ NPM-pakettivarastossa joidenkin pakettien
           ↳ versioissa oli selittämättömästi samat
           ↳ julkaisuajankohdat. Nämä vääristäisivät
           ↳ tilastotietoja, ja siksi ne suodatetaan pois
19             AND vts.flag = 'major_version' -- Vain major-versioita
20     ), helper_query AS (
21         SELECT vts.version_reference FROM version_time_sequence
           ↳ vts LEFT JOIN next_steps ON
           ↳ next_steps.package=vts.package WHERE
           ↳ next_steps.order_sequence=vts.order_sequence+1
22     ) SELECT COUNT(DISTINCT version_reference) FROM helper_query
           ↳ INTO ni_total_versions;
23
24     WHILE ti <= (5*12)
25
26     LOOP
27         RAISE NOTICE 'Current month: %', ti;
```

```

28      DECLARE
29          di bigint;
30          ni bigint;
31      BEGIN
32
33          -- Estimaattorina käytetään tulosarjaa (1 - (di/ni)), jossa:
34          -- di = tietyn ajanhetken (ti) tapahtumien määrä
35          -- ni = niiden yksilöiden määrä, jotka ovat jääneet eloon
36          --   ↪ (ei tapahtumia)
37
38          -- Tässä tapauksessa, di kuvaa niiden versioiden määrää
39          --   ↪ jotka ovat saaneet päivityksen tietyllä ajanjaksolla
40          -- ja ni niiden versioiden määrää jotka eivät ole
41          --   ↪ päivittyneet aiempina ajanjaksoina
42
43          -- Valitse ne versiot jotka ovat päivittyneet tällä
44          --   ↪ ajanhetkellä.
45          -- Tässä tapauksessa siis kaikki edeltävät versiot, joille
46          --   ↪ on olemassa päivitysaskel seuraavaan major-versioon
47      RAISE NOTICE 'Approximating count of updated versions...';
48      WITH next_steps AS (
49          SELECT vts.package, vts.order_sequence
50          FROM version_time_sequence vts
51          WHERE (extract(year from vts.offset_from_major_version)
52                ↪ * 12 +
53                extract(month from
54                       ↪ vts.offset_from_major_version))::bigint = ti
55          AND vts.offset_from_major_version IS NOT NULL
56          AND vts.offset_from_major_version != '0 seconds'
57          AND vts.flag = 'major_version'
58      ), helper_query AS (
59          SELECT vts.version_reference FROM version_time_sequence
60          ↪ vts LEFT JOIN next_steps ON
61          ↪ next_steps.package=vts.package WHERE
62          ↪ next_steps.order_sequence=vts.order_sequence+1
63      )

```

```

54     SELECT COUNT(DISTINCT version_reference) FROM helper_query
      ↪ INTO di;
55 RAISE NOTICE 'di: %', di;
56
57     -- Selvinneiden yksilöiden määrä on päivittymättömien
      ↪ versioiden määrä
58     ni := ni_total_versions;
59
60     RAISE NOTICE 'ni: %', ni;
61     RAISE NOTICE 'Approximately % percentage of versions updated
      ↪ within % months of previous applicable version',
      ↪ (di::double precision / ni::double precision)*100.0, ti;
62     IF ni < (0 :: bigint) THEN
63         RAISE NOTICE 'No surviving versions! Stop!';
64         -- Ei versioita? Lopeta tähän
65         EXIT;
66     end if;
67
68     probability = probability * (1.0 - (di::double precision /
      ↪ ni::double precision));
69     RAISE NOTICE 'Probability now: %', probability;
70     RETURN QUERY SELECT ti, probability;
71
72     -- Vähennä kokonaisversiomäärästä
73     ni_total_versions := ni_total_versions - di;
74     END;
75
76     -- Siirry seuraavaan kuukauteen
77     ti = ti + 1;
78     end loop;
79     end;
80     $$;

```

D 100 yleisintä paketissa tavattua lisenssiä

```
0 license_count,license 35 4386,Apache 2
1 5768342,MIT 36 4098,Apache License 2.0
2 1796292,ISC 37 3772,GPL-3.0+
3 1184020,"" 38 3610,(Apache-2.0 OR MIT)
4 829011,Apache-2.0 39 3533,Apache-2
5 163793,BSD-3-Clause 40 3257,AGPL-3.0-or-later
6 103019,UNLICENSED 41 3218,LicenseRef-LICENSE
7 93148,GPL-3.0 42 3120,"<deprecated object sequence
8 63162,BSD ↪ of multiple licenses, fix
9 58849,EPL-2.0 OR GPL-2.0 WITH ↪ this manually if needed>"
↪ Classpath-exception-2.0 43 3048,LGPL
10 51205,BSD-2-Clause 44 3044,AGPL-3.0-only
11 42181,Commercial 45 2956,SEE LICENSE IN license.txt
12 33840,SEE LICENSE IN LICENSE 46 2906,0BSD
13 29309,AGPL-3.0 47 2747,EPL-1.0
14 28647,MPL-2.0 48 2713,(MIT OR Apache-2.0)
15 27854,Apache 2.0 49 2631,SEE LICENSE IN README.md
16 25263,LGPL-3.0 50 2629,Apache2
17 22631,Unlicense 51 2500,SEE LICENCE IN LICENSE
18 17633,WTFPL 52 2302,GNU
19 15633,SEE LICENSE IN LICENSE.md 53 2301,MIT License
20 14767,GPL-2.0 54 2236,GPL-3.0-only
21 10553,Apache 55 2210,SEE LICENCE in LICENSE
22 10306,EUPL-1.1 56 2179,LGPL-2.1
23 9522,GPL-3.0-or-later 57 2007,GPLv2
24 9485,OSL-3.0 58 1962,GPL-v3
25 9432,CC0-1.0 59 1916,See license in LICENSE.md
26 8792,SEE LICENSE in LICENSE.md 60 1852,SEE LICENSE IN https://chromi
↪ ium.googleusercontent.com/chromium
↪ /src/+/master/LICENSE
27 7726,GPL 61 1816,Artistic-2.0
28 7534,SEE LICENSE IN LICENSE.txt 62 1795,Apache License v2.0
29 7093,GPLv3 63 1757,LGPL-3.0+
30 6997,SEE LICENSE IN license 64 1711,EPL-2.0
31 6708,CC-BY-3.0 65 1585,Proprietary
32 5195,GPL-2.0-or-later 66 1517,GPL-2.0+
33 5194,CC-BY-4.0
34 4424,SEE LICENSE IN EULA.pdf
```


| | | | |
|----|--|-----|---|
| 67 | 1466,GNU GPLv3 | 84 | 1094,AAL |
| 68 | 1414,Copyright 2016 | 85 | 1081,CC-BY-SA-4.0 |
| 69 | 1365,MIT/X11 | 86 | 1076,CC0 |
| 70 | 1314,none | 87 | 1053,LGPLv3 |
| 71 | 1288,private | 88 | 1053,"Apache License, Version 2.0" |
| 72 | 1279,GNU Lesser General Public → License v3.0 | 89 | 995,New BSD |
| 73 | 1277,"MIT[url:String("http://www → .opensource.org/licenses/mit- → license.php")]"]" | 90 | 960,Zlib |
| 74 | 1271,CC-BY-NC-SA-4.0 | 91 | 951,SEE LICENSE IN License.md |
| 75 | 1235,"MIT, EPL" | 92 | 907,SEE LICENSE IN LICENSE.TXT |
| 76 | 1215,UNLICENCED | 93 | 900,MPL 2.0 |
| 77 | 1208,LGPL-3.0-or-later | 94 | 881,MIT AND Apache-2.0 AND → OFL-1.1 AND CC0-1.0 |
| 78 | 1185,pemrouz.mit-license.org | 95 | 869,UNLICENCED |
| 79 | 1143,mit | 96 | 867,Apache Public License v2 |
| 80 | 1120,GPL3 | 97 | 859,Public Domain |
| 81 | 1119,CC-BY-NC-ND-4.0 | 98 | 853,(GPL-2.0 OR LGPL-2.1 OR → MPL-1.1) |
| 82 | 1114,Apache version 2.0 | 99 | 853,Apache License |
| 83 | 1113,AGPL | 100 | 852,GPL v3 |

E 100 suosituinta pakettia suosikkimerkintöjen perusteella

| | | | |
|----|-----------------------------|----|------------------------|
| 0 | favoriting_user_count, name | 36 | 448, redis |
| 1 | 2563, express | 37 | 441, dotenv |
| 2 | 1505, request | 38 | 439, inquirer |
| 3 | 1486, lodash | 39 | 438, winston |
| 4 | 1429, gulp | 40 | 435, browser-sync |
| 5 | 1213, async | 41 | 432, minimist |
| 6 | 1061, pm2 | 42 | 432, shelljs |
| 7 | 1031, moment | 43 | 427, karma |
| 8 | 1011, commander | 44 | 427, morgan |
| 9 | 931, chalk | 45 | 425, fs-extra |
| 10 | 928, browserify | 46 | 425, q |
| 11 | 794, socket.io | 47 | 423, mongodb |
| 12 | 744, react | 48 | 418, npm-check-updates |
| 13 | 741, forever | 49 | 403, yo |
| 14 | 741, mongoose | 50 | 381, express-session |
| 15 | 703, mocha | 51 | 374, npm |
| 16 | 699, debug | 52 | 374, gulp-autoprefixer |
| 17 | 698, grunt | 53 | 365, jquery |
| 18 | 621, nodemon | 54 | 358, cordova |
| 19 | 621, cheerio | 55 | 351, glob |
| 20 | 615, body-parser | 56 | 349, hapi |
| 21 | 598, bluebird | 57 | 345, mkdirp |
| 22 | 591, bower | 58 | 343, gulp-imagemin |
| 23 | 574, underscore | 59 | 341, bcrypt |
| 24 | 554, axios | 60 | 339, sequelize |
| 25 | 549, webpack | 61 | 333, coffee-script |
| 26 | 546, gulp-uglify | 62 | 329, eslint |
| 27 | 525, passport | 63 | 327, less |
| 28 | 520, http-server | 64 | 323, gulp-rename |
| 29 | 511, yargs | 65 | 320, rimraf |
| 30 | 509, nodemailer | 66 | 319, grunt-cli |
| 31 | 491, mysql | 67 | 316, jsonwebtoken |
| 32 | 489, validator | 68 | 316, chai |
| 33 | 478, gulp-sass | 69 | 313, uuid |
| 34 | 457, colors | 70 | 310, bootstrap |
| 35 | 451, gulp-concat | 71 | 308, gulp-sourcemaps |

| | | | |
|----|----------------|-----|--------------------|
| 72 | 308,superagent | 87 | 262,koa |
| 73 | 298,ejs | 88 | 261,node-sass |
| 74 | 297,jade | 89 | 259,react-router |
| 75 | 296,cross-env | 90 | 257,node-schedule |
| 76 | 286,npm-check | 91 | 255,ws |
| 77 | 284,helmet | 92 | 248,redux |
| 78 | 282,del | 93 | 246,connect |
| 79 | 280,xml2js | 94 | 244,formidable |
| 80 | 279,joi | 95 | 239,node-inspector |
| 81 | 278,multer | 96 | 238,cookie-parser |
| 82 | 275,supertest | 97 | 237,angular |
| 83 | 271,faker | 98 | 235,gm |
| 84 | 269,cors | 99 | 233,co |
| 85 | 268,uglify-js | 100 | 232,vue |
| 86 | 263,typescript | | |

F 100 suosituinta avainsanaa pakettiesiintymien perusteella

| | | | |
|----|----------------------------------|----|------------------------|
| 0 | distinct_reference_count,keyword | 36 | 5806,server |
| 1 | 46020,react | 37 | 5680,library |
| 2 | 18655,javascript | 38 | 5598,async |
| 3 | 14899,node | 39 | 5556,gruntplugin |
| 4 | 14642,vue | 40 | 5543,template |
| 5 | 14605,angular | 41 | 5249,file |
| 6 | 14537,react-component | 42 | 5151,font |
| 7 | 14240,api | 43 | 5146,middleware |
| 8 | 13570,cli | 44 | 5115,data |
| 9 | 13412,typescript | 45 | 5102,image |
| 10 | 13016,plugin | 46 | 5079,module |
| 11 | 12912,css | 47 | 5036,ecosystem:cordova |
| 12 | 12278,react-native | 48 | 4967,parser |
| 13 | 10889,component | 49 | 4914,android |
| 14 | 9862,webpack | 50 | 4888,ios |
| 15 | 9348,json | 51 | 4857,config |
| 16 | 8953,yeoman-generator | 52 | 4726,client |
| 17 | 8666,npm | 53 | 4589,markdown |
| 18 | 8665,test | 54 | 4554,stream |
| 19 | 8398,express | 55 | 4405,object |
| 20 | 7904,framework | 56 | 4248,string |
| 21 | 7788,html | 57 | 4243,sass |
| 22 | 7203,eslint | 58 | 4194,cordova-android |
| 23 | 7111,js | 59 | 4162,array |
| 24 | 7073,redux | 60 | 4116,rest |
| 25 | 7069,web | 61 | 4020,package |
| 26 | 6674,nodejs | 62 | 3995,gulpplugin |
| 27 | 6598,browser | 63 | 3959,babel |
| 28 | 6567,ui | 64 | 3954,app |
| 29 | 6497,generator | 65 | 3932,testing |
| 30 | 6380,http | 66 | 3849,log |
| 31 | 6296,ember-addon | 67 | 3781,validation |
| 32 | 6114,promise | 68 | 3638,angular2 |
| 33 | 5931,components | 69 | 3637,aws |
| 34 | 5930,gulp | 70 | 3612,form |
| 35 | 5866,es6 | 71 | 3605,typeface |

| | | | |
|----|-------------------|-----|---------------------|
| 72 | 3594,font family | 87 | 3049,functional |
| 73 | 3589,utility | 88 | 3026,input |
| 74 | 3568,cordova | 89 | 2983,utils |
| 75 | 3477,loader | 90 | 2977,text |
| 76 | 3435,google fonts | 91 | 2965,mongodb |
| 77 | 3433,bootstrap | 92 | 2949,map |
| 78 | 3422,router | 93 | 2928,koa |
| 79 | 3419,reactjs | 94 | 2924,request |
| 80 | 3413,database | 95 | 2912,github |
| 81 | 3366,cordova-ios | 96 | 2909,parse |
| 82 | 3329,svg | 97 | 2909,google |
| 83 | 3299,cache | 98 | 2905,mobile |
| 84 | 3202,native | 99 | 2866,util |
| 85 | 3183,git | 100 | 2865,authentication |
| 86 | 3118,jquery | | |

G 100 eniten riiputuinta pakettiversiota

| | | | |
|----|-----------------------------|----|----------------------------------|
| 0 | version_specifier,count | 36 | react-dom@15.6.2,94321 |
| 1 | lodash@4.17.15,834105 | 37 | redux@3.7.2,91295 |
| 2 | prop-types@15.7.2,436630 | 38 | async@2.6.3,88011 |
| 3 | moment@2.24.0,395918 | 39 | babel-preset-es2015@6.24.1,81889 |
| 4 | tslib@1.10.0,375019 | 40 | webpack@4.41.2,79009 |
| 5 | request@2.88.0,336359 | 41 | rxjs@5.5.12,78628 |
| 6 | classnames@2.2.6,323610 | 42 | node-sass@4.12.0,77523 |
| 7 | express@4.17.1,296679 | 43 | debug@3.2.6,73452 |
| 8 | react@16.10.2,291755 | 44 | react-router-dom@4.3.1,70861 |
| 9 | commander@2.20.3,283240 | 45 | react-redux@5.1.2,70651 |
| 10 | chalk@2.4.2,282455 | 46 | reflect-metadata@0.1.13,70645 |
| 11 | mkdirp@0.5.1,215696 | 47 | aws-sdk@2.553.0,70357 |
| 12 | react-dom@16.10.2,215529 | 48 | debug@4.1.1,69916 |
| 13 | uuid@3.3.3,201194 | 49 | gulp@3.9.1,69806 |
| 14 | bluebird@3.7.1,188813 | 50 | isomorphic-fetch@2.2.1,66720 |
| 15 | chalk@1.1.3,183027 | 51 | @babel/core@7.6.4,66106 |
| 16 | vue@2.6.10,179748 | 52 | q@1.5.1,65689 |
| 17 | babel-runtime@6.26.0,178229 | 53 | babel-preset-react@6.24.1,65148 |
| 18 | body-parser@1.19.0,155047 | 54 | object-assign@4.1.1,63151 |
| 19 | @babel/runtime@7.6.3,152943 | 55 | inquirer@6.5.2,62965 |
| 20 | glob@7.1.4,151662 | 56 | qs@6.9.0,62092 |
| 21 | minimist@1.2.0,139687 | 57 | babel-preset-env@1.7.0,60393 |
| 22 | axios@0.18.1,137100 | 58 | typescript@3.6.4,59823 |
| 23 | underscore@1.9.1,127147 | 59 | node-fetch@2.6.0,59655 |
| 24 | react@15.6.2,126500 | 60 | immutable@3.8.2,59098 |
| 25 | colors@1.4.0,122392 | 61 | path@0.12.7,57484 |
| 26 | debug@2.6.9,119656 | 62 | rxjs@6.5.3,56469 |
| 27 | babel-core@6.26.3,117285 | 63 | cookie-parser@1.4.4,56275 |
| 28 | core-js@2.6.10,117194 | 64 | cors@2.8.5,55742 |
| 29 | lodash@3.10.1,111595 | 65 | async@1.5.2,54627 |
| 30 | semver@5.7.1,103778 | 66 | node-uuid@1.4.8,54002 |
| 31 | js-yaml@3.13.1,103243 | 67 | ejs@2.7.1,52808 |
| 32 | rimraf@2.7.1,101748 | 68 | handlebars@4.4.3,52402 |
| 33 | babel-polyfill@6.26.0,98981 | 69 | lodash@2.4.2,52258 |
| 34 | jquery@3.4.1,96240 | 70 | @babel/preset-env@7.6.3,51517 |
| 35 | fs-extra@7.0.1,96030 | 71 | fs-extra@8.1.0,51409 |

72 axios@0.19.0,51352
73 compression@1.7.4,51049
74 winston@2.4.4,50440
75 xml2js@0.4.22,50377
76 vue-router@3.1.3,49970
77 gulp-util@3.0.8,49763
78 zone.js@0.8.29,49684
79 eslint-plugin-import@2.18.2,48771
80 babel-loader@7.1.5,48691
81 jsonwebtoken@8.5.1,48283
82 fs-extra@4.0.3,47921
83 shelljs@0.8.3,47907
84 css-loader@0.28.11,47445
85 through2@2.0.5,47335
86 minimatch@3.0.4,47216
87 redux@4.0.4,46722
88 redux-thunk@2.3.0,46716
89 request-promise@4.2.4,46569
90 co@4.6.0,46505
91 @types/lodash@4.14.144,46187
92 typescript@2.9.2,45966
93 morgan@1.9.1,45729
94 babel-loader@8.0.6,45614
95 async@0.9.2,45447
96 babel-plugin-transform-runtime@6.11.6,
 ↪ 23.0,45362
97 font-awesome@4.7.0,45321
98 babel-cli@6.26.0,44251
99 fs-extra@5.0.0,43335
100 prettier@1.18.2,43325