Re-engineering the Hazard Calculation: toshi-hazard-post and toshi-hazard-store v2

Chris DiCaprio and Chris Chamberlain

Te Tauira Matapae Pūmate Rū i Aotearoa
NSHM The New Zealand National Seismic Hazard Model
A GNS Science Led Research Programme

E mahi ana me
In collaboration with

Ngā hoa tuku pūtea
Funding partners

MINISTRY OF BUSINESS, INNOVATION & EMPLOYMENT
HĪKINA WHAKATUTUKI

Toka Tū Ake EQC

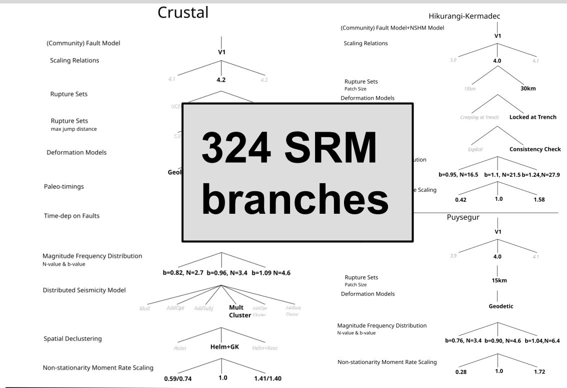
Motivation

`toshi-hazard-store` and `toshi-hazard-post` facilitate calculation of large logic trees

Issues

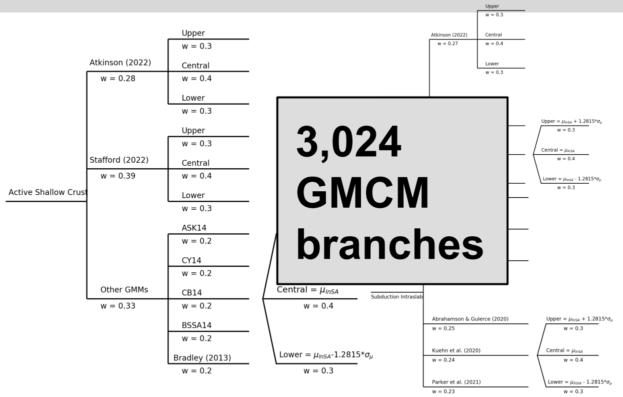
- **Development was done under considerable time pressure**
  - We used tools already used by project and at our disposal: ToshiAPI, dynamoDB
  - Significant technical debt incurred: “it ain’t pretty, but it works”
- **Not user friendly:**
  - Burdensome workflow
  - Requires mimicking NSHM project’s IT stack; not realistic for non-GNS users
- **Performance left on the table**

Refresh: Hazard Calculation



324 SRM branches

✖



3,024 GMCM branches

=

979,776 total branches

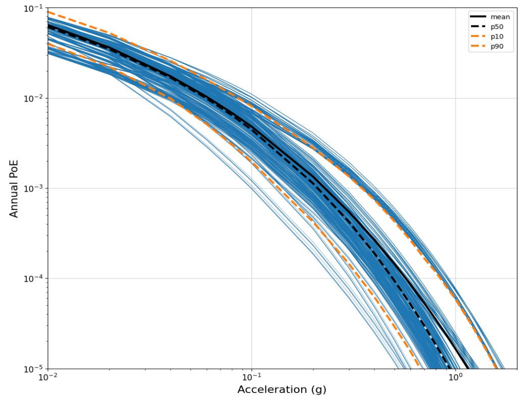
Decompose the model and break calculation into 2 stages

Stage 1

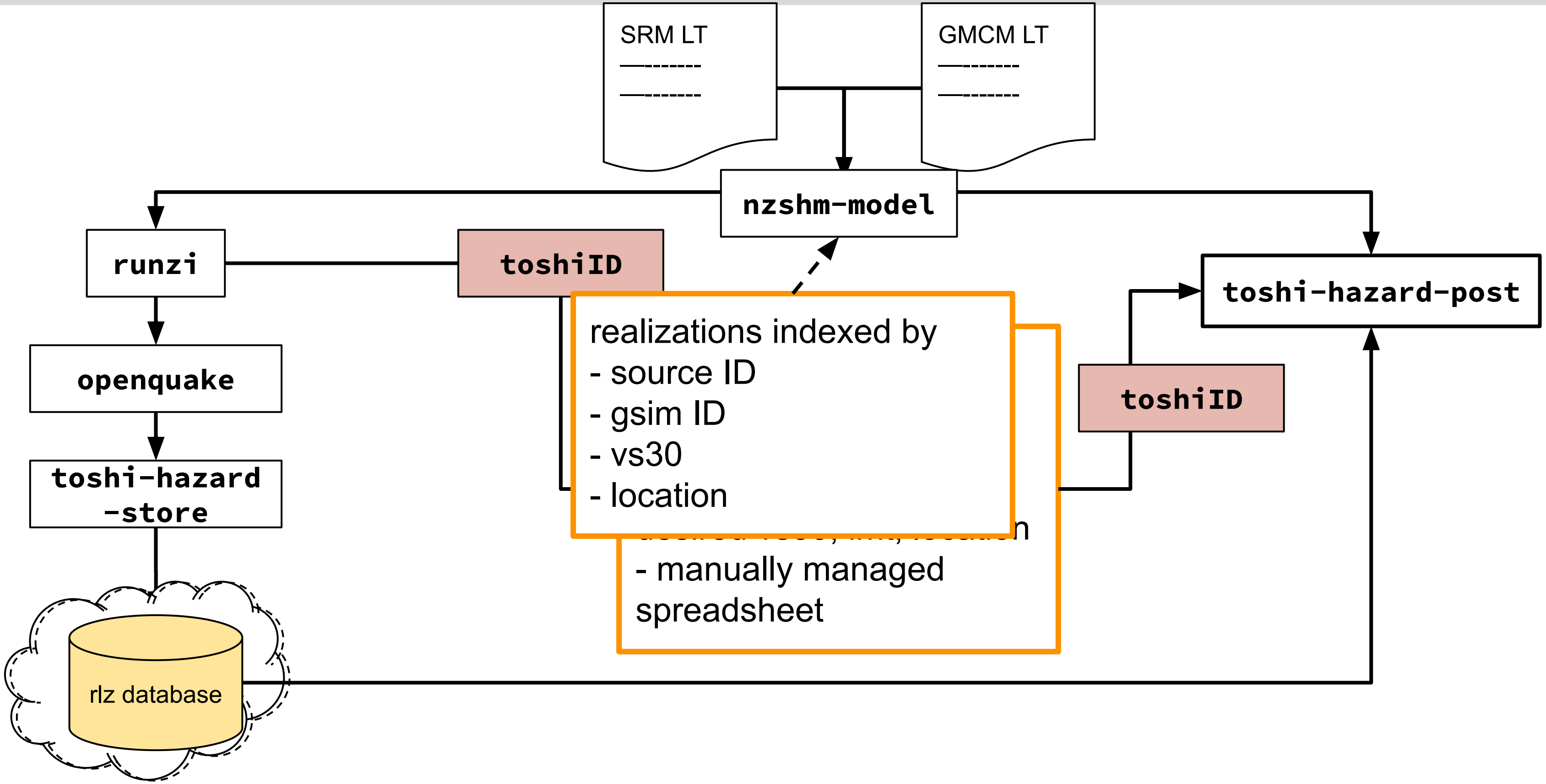
- Calculate hazard for independent components concurrently
- Store **component realizations**

Stage 2

- Combine independent components to form 979,776 realizations
- Calculate aggregate statistics (e.g. weighted mean and fractiles)



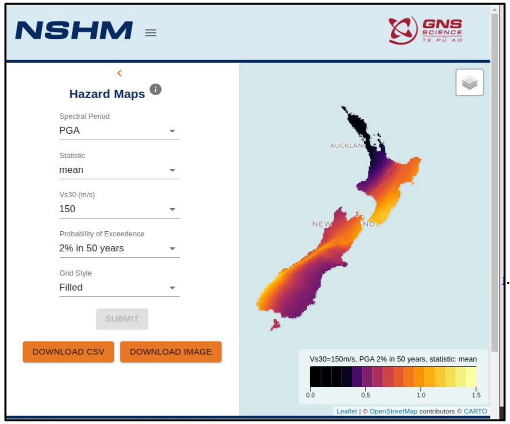
Hazard Workflow: Old vs New



realizations indexed by
- source ID
- gsim ID
- vs30
- location

- manually managed spreadsheet

Calculating and consuming Hazard Curves



Hazard Maps

Spectral Period: PGA

Statistic: mean

Vs30 (m/s): 150

Probability of Exceedence: 2% in 50 years

Grid Style: Filled

SUBMIT

DOWNLOAD CSV

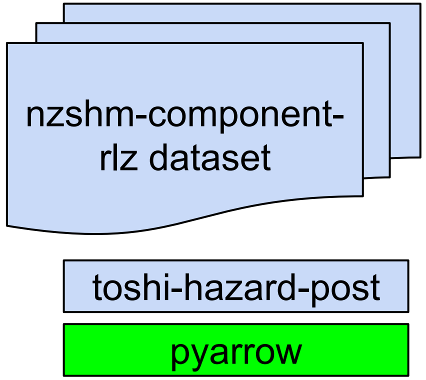
DOWNLOAD IMAGE

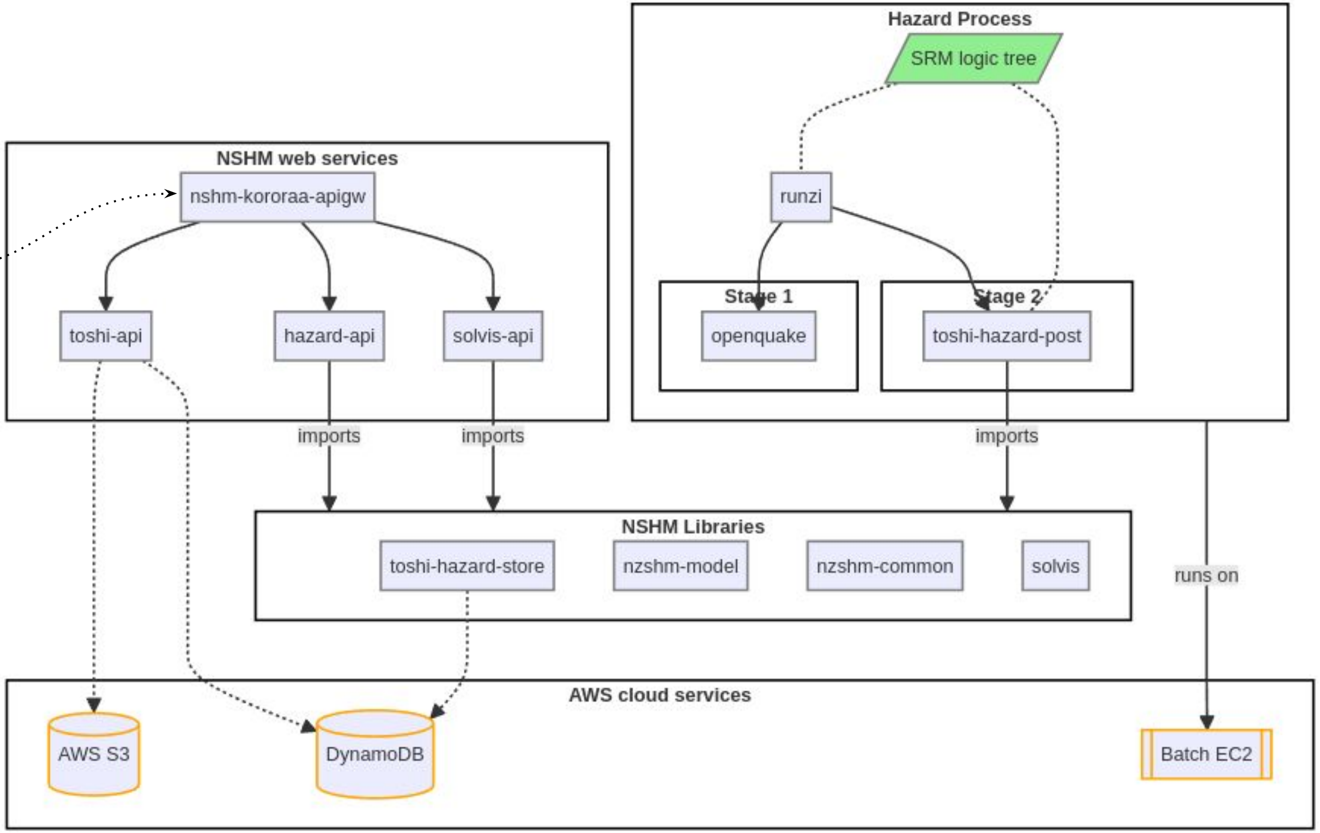
Vs30=150m/s, PGA 2% in 50 years, statistic: mean

0.0 0.5 1.0 1.5

Leaflet | © OpenStreetMap contributors © CARTO

Kororā Web Application





NSHM 2022 = the PSHA branch realisations rodeo

With:

3991 sites,

20 vs30 values,

27 Intensity Measure Types (IMTS) PGA, SA(0.5) etc

~900 source/gsim model permutations

=> 1,965,487,680 individual realisation curves, of 44 points each.

In DynamoDB, approx 2 Terabytes and 200 million objects

In Parquet, approx 0.6 Terabyte and 2 billion rows

Database Types

- **DynamoDB** *still used for website*
- **Sqlite3** *an available anywhere sql DB,*
- **introducing Arrow** *https://arrow.apache.org/docs/python/install.html*

Small demo:

- **reading the NSHM dataset just using pyarrow/S3.**
- **Tables, filtering, to_pandas(), shape**
- **Exploring the data**
- **Identifying branches**

The NSHM realisation dataset schema

```
schema = pa.schema(
    [
        ("compatible_calc_fk", dict_type),   # id for calculation equivalence, for PSHA engine interoperability
        ("calculation_id", dict_type),       # a reference to the original calculation that produced this item
        ("nloc_001", dict_type),             # the location string to three places e.g. "-38.330~17.550"
        ("nloc_0", dict_type),               # the location string to zero places e.g.  "-38.0~17.0")
        ('imt', dict_type),                  # the IMT label e.g. 'PGA', 'SA(5.0)'
        ('vs30', vs30_type),                 # the vs30 value e.g 400
        ('rlz', dict_type),                  # the rlz id from the the original calculation eg “rlz-001”
        ('sources_digest', dict_type),       # a unique hash id for the NSHM LTB source branch
        ('gmms_digest', dict_type),          # a unique hash id for the NSHM LTB gsim branch
        ("values", values_type),             # a list of the 44 IMTL values
    ]
)
```

Introducing nzshm-component-rlz-dataset (from June 2024)

A collection of standardised PSHA realizations to facilitate research and model development, allows for:

- create new models from existing branch realisations (recombine, reweight)
- add new realisations, compare, merge into new models
- make detail comparisons to NSHM_v1.0.4 baseline.

Accessible:

- public access with option to download dataset OR query directly against the cloud store (S3 bucket)
- uses lightweight and simple python libraries
- Productive using only workstation compute - no server-side resources

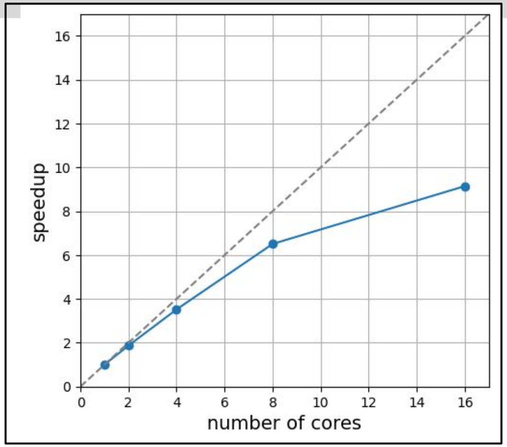
toshi-hazard-post

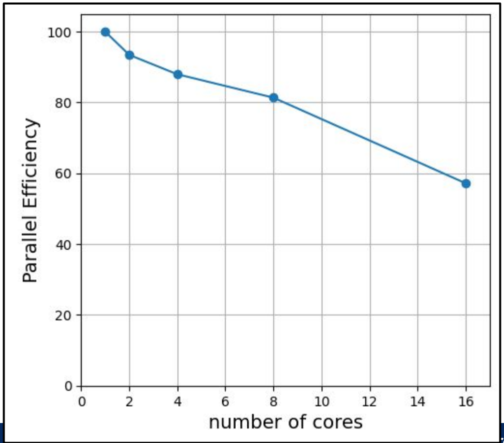
Simplification: 3,485 down to 1,077 lines of code

- Supporting libraries nzhsm-model and nzshm-common
- Removal of toshiAPI dependency

Performance Improvement: 3x faster

- Makes better use of numpy vectorization
- Simplification of logic





36 Core, 64GB workstation

Possible shell demos of libraries?

- common
- model
- solvis

ALL DONE!