InaSAFE Version 4.0 Architecture

Input Layers

Vector Layer ...or... Raster Layer — Exposure

Exposure Layer
- currency_abbreviation
- feature_type

Vector Layer optionally with some selected features ...or... Bounding Box — Aggregation

Aggregation layer provided? No → Convert BBOX to polygon layer with keywords

Yes → Aggregation polygon layer
- aggregation_id
- aggregation_name
- ratios

Keyword change: update field references using InaSAFE preferred field names

Vector Layer ...or... Raster Layer — Hazard

Hazard Layer
- hazard_id
- hazard_class
- hazard_value

Options

Classifications (provides thresholds for continuous data)

Demographic Defaults | Value Maps (for classified data)

Profile | IF Thresholds | Minimum Needs Profile | IF Buffer Zones

Workflow Planner

Exposure is raster population? Yes → Hazard is raster earthquake? No

No

Hazard is raster earthquakes? Yes → Exposure is raster population? Yes

No

No

Raster on Raster Damage Curve

Exposure Layer (raster population)

Aggregation polygon layer
- aggregation_id
- aggregation_name
- ratios

Hazard Layer (raster MMI earthquake)

Prepare Exposure
Resample, clip to aggregation and align to same origin as hazard

Add columns to aggregation polygon layer for:
* mmi_1_affected
* mmi_2_affected
* mmi_3_affected
* mmi_12_affected
* mmi_1_displaced
* mmi_2_displaced
* mmi_3_displaced
...
mmi_12_displaced
* mmi_1_killed
* mmi_2_killed
* mmi_3_killed
* mmi_12_killed
* fatality_rate

Prepare Hazard
Resample, clip to aggregation and align to same origin as exposure

Prepared Exposure Layer (raster population)

Prepared Hazard Layer (raster MMI earthquake)

Loop over every aggregation area

Prepare output affected people layer by making a copy of exposure array layer as numpy array and writing NaNs to it

Burn all cells outside current aggregation area to produce Population in aggregation area.

Burn all cells outside current aggregation area to produce Hazard in aggregation area.

Execute preferred earthquake impact function as defined in user profile.

Superimpose population affected onto output numpy array.

Add row to results layer with aggregation geometry, displaced, affected, fatalities etc. for aggregation area.

Write affected population numpy array to tiff

aggregation_layer created

Earthquake Raster

Population Raster

Exposure Preparation

Field remapping
* copy layer
* exposure_id
* exposure_value
* remove columns

Raster exposure? Yes → Continuous? Yes → Densities — No / Yes

Raster landcover classes

Create fields from defaults
e.g. gender ratio

Output fields:
* exposure_id
* exposure_class

Assign classes based on value_map

User:
* 1 = rice
* 2 = wheat
* 3 = maize
* 4 = urban

InaSAFE:
* agricultural
* urban

Classified exposure with keywords

Features divisible? Yes → Clip exposure with aggregate hazard

Building points
Places
Landcover
Population polygons
Roads

No

Buildings polygons

Exlude all polygons not intersecting aggregate hazard
Smart clip, spatial join ?

Polygonise classified raster hazard
Assign class names based on class ID
Contiguous areas

Calculate counts per cell

We need to describe the mechanism for doing these class mappings

Do we remap these to InaSAFE' terms? Possibly second round of development for this?

For now we deal with this the same way as with hazard classified rasters....

Has absolute data ? No → Yes

Recalculate population counts based on new polygons size (population = new area / old area * old population)

Prepared Exposure Layer

Aggregation Preparation

Field remapping
* copy layer
* use selected features only
* aggregation_id
* aggregation_name
* gender_ratio
* age_ratio
* etc.

Project to CRS of Exposure

Analysis layer is a single feature (can be multipart) that defines the outer perimeter of the analysis area. Its attributes will hold the totals for the entire analysis.

Create analysis layer

Create fields from defaults
e.g. gender ratio

Aggregate Hazard Preparation

Clip hazard to aggregation

input 2: aggregation layer

input 1: hazard layer clipped by aggregation

Union hazard polygons with aggregation areas

Keyword change:
1) change purpose to aggregate_hazard
2) set the type attribute

Output fields:
* aggregation_id
* aggregation_name
* hazard_id (can be NULL)
* hazard_class_name (can be NULL)

Aggregate hazard with keywords

Hazard Preparation

User:
* 1
* 2
* 3
* 4

InaSAFE:
* low
* medium
* high

(from definitions.py)

Copy layer to a local file — No / Vector Hazard? — Yes → Create fields from defaults
e.g. gender ratio

User:
* 1 = moderate
* 2 = high
* 3 = very high
* 4 = total

This is an open question as to whether we use one approach always or select the approach based on exposure data. Contiguous areas make more 'natural' looking hazard areas but are more CPU intensive for processing. Pixel based areas are more CPU efficient but suffer from poor visualisation and the Aggregate Hazard area will have high geometric complexity.

Continuous Raster? Yes → Classify continuous raster hazard (e.g. flood depth)

Keyword change: continuous to classified

No → Polygonise classified raster hazard
Assign class names based on class ID
Pixel based | Contiguous areas

Field remapping
* copy layer
* hazard_id
* hazard_value
* remove columns

Keyword change: data_type changes to classified

Classify Continuous Hazard (e.g. flood depth) and assign class names
Assign class names based on class ID

Keyword change: attribute field set to type

Continous Vector? Yes → Polygon Hazard? — No / Yes

No

Keyword change: Use value map to map dataset classes to InaSAFE hazard concept

or (post v4.0):

Run analysis and report based on users classes? (in a future version we could look at this, We would also need to make the user specify which classes represent affected and which don't).

Assign classes based on value_map

Keyword change: attribute_field set to type

Keyword change: none (geometry type is dynamically read)

Need buffering (e.g. volcano point)

Keyword change: none

Project to CRS of Exposure

Prepared Hazard Layer Classified polygon hazard with keywords

Output fields:
* hazard_id
* hazard_class_name

Combination of exposure with aggregate hazard

Exposure Vector — Yes / No

Exposure Features divisible? No / Yes

Highest class of hazard is assigned when more than one overlaps

Indivisible Polygons

Union exposure features to the aggregate hazard

Point | Line | Divisible Polygon

Raster

Zonal stats raster exposure on vector aggregate hazard

Output fields:
* exposure_id
* exposure_class
* aggregation_id
* aggregation_name
* hazard_id (can be NULL)
* hazard_class (can be NULL)

Intermediate Impact Layer

Post process intermediate impact layer

Post process intermediate impact layer

Output fields:
* exposure_id
* exposure_class
* aggregation_id
* aggregation_name
* hazard_id (can be NULL)
* hazard_class (can be NULL)
* extra fields from post processors

Impact Layer

Updated Aggregate hazard with keywords and post processor results

aggregate hazard (produced from zonal stats)

Summary Calculation

impact layer

aggregate hazard layer

Aggregate summary

Number of features per aggregate hazard area will be the **unique combinations of hazard id and aggregation id**.

Ratio fields will be dropped.

Absolute value fields (e.g. count fields like pop) will be summed.

String fields will be dropped (e.g. building type)

Fields will be added with **totals** for all features per aggregate hazard area and totals for each exposure class (e.g. house, school etc.).

Output fields:
* aggregation_id
* aggregation_name
* hazard_id (can be NULL)
* hazard_class (can be NULL)
* count for each exposure feature per aggregation and hazard poly.

Aggregate Hazard Results

aggregation layer

Aggregate summary

aggregated earthquake impact layer

Number of features per aggregate area will be calculated using the **unique aggregation ids**.

Ratio fields will be dropped.

Absolute value fields (e.g. count fields like pop) will be summed.

String fields will be dropped (e.g. building type)

If needed, fields will be added with **totals** for all features per aggregate hazard area and totals for each exposure class (e.g. house, school etc.).

Run Postprocessors
Minimum needs for raster population

Output fields:
* aggregation_id
* aggregation_name
* totals

Aggregation Results

Table will have only one row (for whole study area).

Ratio fields will be dropped.

Absolute value fields (e.g. count fields like pop) will be summed.

String fields will be dropped (e.g. building type)

If needed, fields will be added with **totals** for all features per aggregate hazard area and totals for each exposure class (e.g. house, school etc.).

analysis layer

Aggregate summary

Output fields:
* entire area
* totals

Analysis Layer

Post-Processor

Identify input layer fields

Get list of all post processors

Post processing should be written to aggregate-hazard dataset (e.g. number of females in each area) and to the exposure dataset (e.g. cost per building)

Filter post processors based on available input layer fields

Loop through post processors writing new output fields to input layer and adding keywords to input layer metadata

Updated layer

Example Post Processors

* class rate
* class value
* size rate
* value
* minimum needs
* female count
* youth count
* adult count
* elderly count
* disabled count
* building vulnerability
* etc.

Comments from Charlotte:

Be sure to leave original datasets untouched.

Reports should still show e.g. input data was continuous hazard, cell size etc

Where does user get told that their data was turned into polygons.

Add message explaining the step name to progress callbacks.

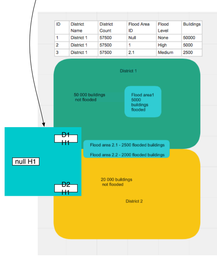
For Future?

Should we be using UNION rather here? We keep everything and then report only on those areas that have an aggregation id.
* get rid of clip step
* if hazard if bigger than exposure we will keep the non overlapped areas

Perhaps as a second path based on options - in some cases user may have large hazard dataset but only want reporting for a specific area in which case we may want to stick with the clip and mask approach...

The risk with this approach is that exposure data may be huge beyond the aggregation area. This idea is on hold for now.

ID	District Name	District Count	Flood Area ID	Flood Level	Buildings
1	District 1	57500	Null	None	50000
2	District 1	57500	1	High	5000
3	District 1	57500	2.1	Medium	2500



Reporting

Derived Hazard Layer

Reference to Aggregation Layer

Aggregate Hazard Layer

Impact Layer

Impact Metadata
- Povenance
- ISO 19115 Mandatory Fields
- InaSAFE Keywords

Deferred Output Generator

JSON Document
- Provenance
- Actions & Keywords
- Impact Geometries
- Impact Table
- Aggregate Hazard Geometries
- Aggregate Hazard Table

Report Filter will use metadata in templates & styles and metadata in impact and aggregate hazard layer to determine which styles and templates can be used

Presentation Logic
Report Filter

Standards

e.g. This template needs age and types and whether it works on the impact table or aggregate hazard table.

Reports for different purposes:

*Preparedness
* Response
* Contingency planning

Aggregate Hazard Table
- Template Metadata — Template 1
- Template Metadata — Template 2
- Template Metadata — Template ... n

Cartographic Styling
- Style 1 — Template Metadata
- Style 2 — Template Metadata
- Style ... n — Template Metadata

e.g. This template needs age and types and whether it works on the impact table or aggregate hazard table.

Map Reports

Ultra Mega Flux Capacitor

Renderer: stack bar, bar, chart, pie, donut, table

Expected Fields: e.g. Population, Female count

Available Fields: e.g. Population, Female count, Youth Count, Adult Count, Elderly Count

Inputs: Table, Featrure Fields, Flat Table, Message, Dict, etc.