Large-scale EXecution for Industry & Society

Deliverable D6.3

D6.3 Pilot improvements: Evaluation of Software Development

Co-funded by the Horizon 2020 Framework Programme of the European Union
Grant Agreement Number 825532
ICT-11-2018-2019 (IA - Innovation Action)

DELIVERABLE ID | TITLE	D6.3|CEA — Pilot improvements: Evaluation of Software Development
RESPONSIBLE AUTHOR	Stéphane Louise (CEA)
WORKPACKAGE ID | TITLE	WP6| Earthquake and Tsunami large scale pilot
WORKPACKAGE LEADER	CEA
DATE OF DELIVERY (CONTRACTUAL)	31/09/2021 (M33)
DATE OF DELIVERY (SUBMITTED)	30/11/2021 (M35)
VERSION | STATUS	V1.3 | **Final**
TYPE OF DELIVERABLE	R (Report)
DISSEMINATION LEVEL	PU (Public)
AUTHORS (PARTNER)	Stéphane Louise (CEA); Danijel Scholremmer (GFZ); Natalja Rakowsky (AWI); Andrea Ajmar (ITHACA); Lorenza Bovio (ITHACA)
INTERNAL REVIEW	Tomáš Martinovič (IT4I); Donato Magarielli (Avio Aero)

Project Coordinator: Dr. Jan Martinovič – IT4Innovations, VSB – Technical University of Ostrava
E-mail: jan.martinovic@vsb.cz, **Phone:** +420 597 329 598, **Web:** https://lexis-project.eu

DOCUMENT VERSION

VERSION	MODIFICATION(S)	DATE	AUTHOR(S)
0.1	Creation and table of content	03/10/2021	Stéphane Louise (CEA)
0.2	Description of TsunAWI improvements	04/10/2021	Natalja Rakowsky (AWI); Sven Harig (AWI)
0.3	Added model of computation and workflow description	06/10/2021	Stéphane Louise (CEA)
0.4	Open Building Maps updates, Quadtree and Rabotnik sections	08/10/2021	Danijel Scholremmer (GFZ)
0.5	Satellite Emergency Mapping definition	10/10/2021	Lorenza Bovio (ITHACA)
0.6	First version for review	11/10/2021	Stéphane Louise (CEA); Natalja Rakowsky (AWI)
0.7	Updated Satellite Emergency Mapping section based on the comments from reviewers	20/10/2021	Lorenza Bovio (ITHACA)
0.8	Addressing of the comments from reviewers, mainly in TsunAWI workflow task description and Open Building Maps updates section	22/10/2021	Stéphane Louise (CEA)
0.9	Updated measurements of TsunAWI runs on Karolina cluster, minor improvements towards final version	25/10/2021	Stéphane Louise (CEA); Natalja Rakowsky (AWI)
1.0	Final check of the deliverable	31/10/2021	Kateřina Slaninová (IT4I)
1.1	Minor updates of the content	12/11/2021	Danijel Scholremmer (GFZ)
1.2	Update according to the comments from the final check	16/11/2021	Stéphane Louise (CEA)
1.3	Final check of the deliverable	30/11/2021	Kateřina Slaninová; Jan Martinovič (IT4I)

GLOSSARY

ACRONYM	DESCRIPTION
AOI	Area of Interest
BPMN	Business Process Model Notation: a graphical notation to describe and model business processes.
CEMS	Copernicus Emergency Management Service
DDI	Distributed Data Infrastructure
EQ	Earthquake
FEP	First Estimate Product
GIS	Geographical Information System: a database and additional functions for data with a geographical meaning and coordinates.
MOC	Model of Computation
QML	The name of an actor in the MoC producing a QuakeML file
QUAKEML	Quake Markup Language: a flexible, extensible and modular XML representation of seismological data, e.g. epicentre, hypocentre, magnitude.
RISE	Project in Horizon 2020: “Real-time earthquake rIsk reduction for a reSilient Europe”
RM	Rapid Mapping
SDF	Synchronous Data Flow
SEM	Satellite-based Emergency Mapping: the process of using remote sensing data to produce maps for emergencies
TSUNAWI	Tsunami simulation code developed at AWI

TABLE OF PARTNERS

ACRONYM	PARTNER
Avio Aero	GE AVIO SRL
Atos	BULL SAS
AWI	ALFRED WEGENER INSTITUT HELMHOLTZ ZENTRUM FUR POLAR UND MEERESFORSCHUNG
BLABS	BAYNCORE LABS LIMITED
CEA	COMMISSARIAT A L ENERGIE ATOMIQUE ET AUX ENERGIES ALTERNATIVES
CIMA	CENTRO INTERNAZIONALE IN MONITORAGGIO AMBIENTALE - FONDAZIONE CIMA
CYC	CYCLOPS LABS GMBH
ECMWF	EUROPEAN CENTRE FOR MEDIUM-RANGE WEATHER FORECASTS
EURAXENT	MARC DERQUENNES
GFZ	HELMHOLTZ ZENTRUM POTSDAM DEUTSCHESGEOFORSCHUNGSZENTRUM GFZ
ICHEC	NATIONAL UNIVERSITY OF IRELAND GALWAY / Irish Centre for High-End Computing
IT4I	VYSOKA SKOLA BANSKA - TECHNICKA UNIVERZITA OSTRAVA / IT4Innovations National Supercomputing Centre
ITHACA	ASSOCIAZIONE ITHACA
LINKS	FONDAZIONE LINKS / ISTITUTO SUPERIORE MARIO BOELLA ISMB
LRZ	BAYERISCHE AKADEMIE DER WISSENSCHAFTEN / Leibniz Rechenzentrum der BAdW
NUM	NUMTECH
O24	OUTPOST 24 FRANCE
TESEO	TESEO SPA TECNOLOGIE E SISTEMI ELETTRONICI ED OTTICI

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

EXECUTIVE SUMMARY

This deliverable describes the analysis, evolution, and planned evolutions of the main software components of the earthquake and tsunami large scale pilot of the LEXIS project. This pilot is a use case mixing up Cloud, Big Data and high performance computing (HPC) components chained together in a flow with real-time constraints, and exercise the federation of resources and technologies at the convergence of cloud, big data and high performance computing that the LEXIS project is building.

Position of the deliverable in the whole project context

The LEXIS project relies on three pilots to validate and deploy its technology and infrastructure advances, giving to each pilot its own work package. The work package 6 (WP6) is dedicated to the earthquake and tsunami large scale pilot. After the first Task 6.1 of scenarios and baseline requirements definition, this work package is running Task 6.2, a development task where individual components of the pilot are improved to benefit from the technology of the project and to prepare them for integration on the LEXIS platform, integration is effectively undertaken in Task 6.3. This deliverable summarizes changes in the software that is used in the earthquake and tsunami large scale pilot. It also contains the requirements of this pilot on the orchestration tools and LEXIS Distributed Data Infrastructure and how they were handled in the WP6.

Description of the deliverable

This deliverable contains an updated description of the tsunami and earthquake large scale pilot workflow, in particular, the scheduling groups in the workflow, with a precise and detailed version that show how some level of resilience and guarantees are achieved. It also describes how the work done on the model of computation adaptation allowed better coordination and merge of the data-paths. Globally, on the base of the LEXIS platform, analysis, and evolution of the various components of the pilot were performed to the use-cases of WP6. Among those, the OpenBuildingMap update process to the computation of exposure data, the computation of landmass per tile (or global tile-based coastline dataset), the computation of building completeness per tile, the ShakeMap generation, and in particular the rapid loss assessment computations (aggregate and building-wise), in parallel the fast simulation of tsunami propagation and inundation is done with TsunAWI on several instances, and then the composite dataset is utilized, using a methodology to evaluate the gains of the pilot on the satellite-based emergency mapping process.

1 INTRODUCTION

The LEXIS project is developing solutions for federated HPC and Cloud resources, enhanced by burst buffer technology. This is comprised of three pillars, that is infrastructure, orchestration and data management. The project relies on three pilots representing different domains and exercising the LEXIS platform.

1.1 THE GOAL OF THE PROJECT

The pilot objective is to demonstrate a time constrained workflow for disasters, combining earthquake loss assessment and tsunami simulations into a flow targeting on-time availability of first estimates and detailed estimates, so as to support emergency response decisions and enhance the production of emergency mapping services.

It combines four main application components: A ShakeMap [1] code from the OpenQuake library [2] to compute ground-motion distributions for earthquakes; TsunAWI [1], a code simulating the propagation and inundation of an earthquake-triggered tsunami; OpenBuildingMap [1], a database of geographical information focusing on the built environment and an associated building classification and loss assessment method allowing detailed damage assessments at the building scale after an earthquake and tsunami event, and a satellite-based emergency mapping process producing post-disaster map products. Those four applications are integrated in a workflow described by the PolyGraph [3] model of computation (MoC), and implemented over the infrastructure, orchestration and data management of the LEXIS platform.

The pilot is based on the notion of providing timely results in an emergency situation triggered by a disaster event, a strong effort is made to ensure that early estimates can be provided in a very short time by optimising each of the relevant component and finding innovative implementation solutions (aggregation, suitable indexing schemes), as well as ensuring via the (MoC) that the orchestration of the pilot can be organized so as to respect deadlines. Additional reference datasets are considered, in particular, to allow for a better understanding of the gains offered by this pilot compared to previous implementations of the production of emergency mapping services, and consideration is taken onto the resources needed to make this pilot sustainable in the long term.

2 EVOLUTION OF THE WORKFLOW

As an essential component of the pilot, the workflow has been confirmed and mainly only refined on very precise aspects since the previous update as documented in Deliverable D6.2 [4]. The large scale picture and design of the workflow still conform to the structure and interactions depicted in Figure 1. Nonetheless, the refinements will be shown in subsections 2.2 and after.

2.1 HIGH-LEVEL OVERVIEW

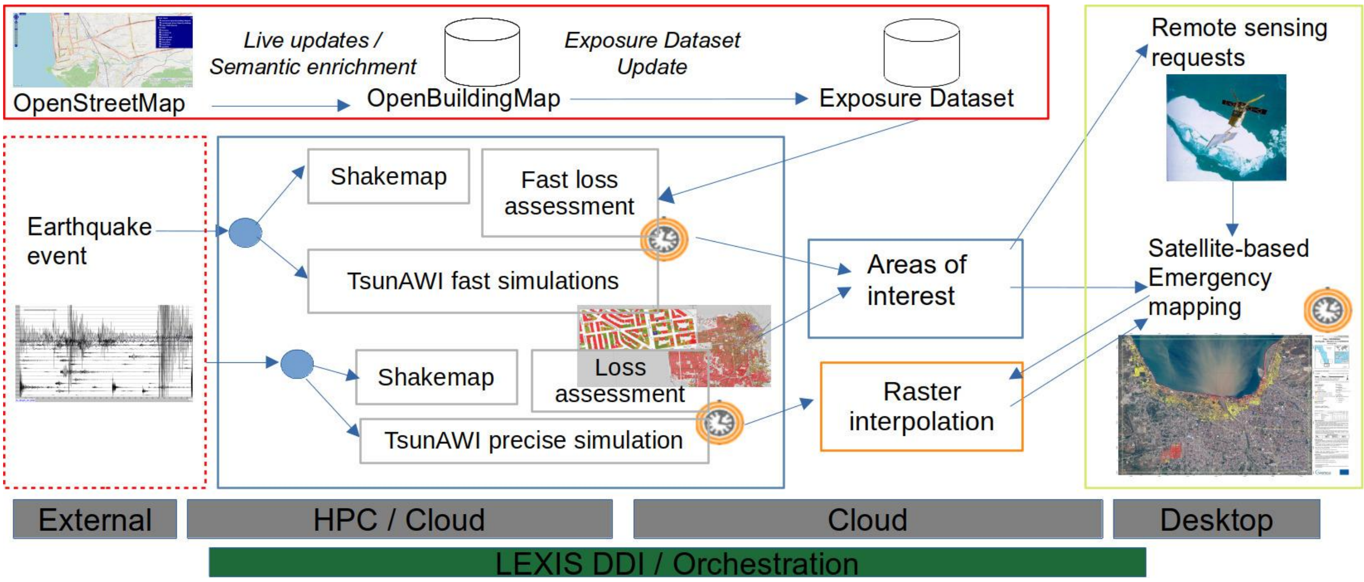

Figure 1 The complete earthquake and tsunami workflow overlaid over the LEXIS orchestration and data infrastructure.
Within red the permanently running exposure dataset creation, in dashed line the event triggered part followed by the compute block (blue), driving in turn the satellite-based emergency mapping

Figure 1 above shows the outlines of the workflow as it was designed. It is composed of multiple blocks, running at different time scales and overlaid over the LEXIS infrastructure. It encompasses the update of an exposure global dataset, an event-triggered tsunami inundation and earthquake loss assessment workflow, and a satellite-based emergency mapping part.

The current aspirational goal as a premise is to have a permanently running process that updates incrementally from the OpenStreetMap world cartographic database, at a minute or so rhythm. From that database, at regular interval, a global exposure dataset is released and uploaded to the LEXIS distributed data infrastructure (DDI).

Whilst this goal is not yet achieved, the GFZ renewed infrastructure and architecture of the update of database can at least let us see a clear path to this achievement. In the meantime, a snapshot of the required database will be provided as an intermediate and working stepping-stone.

Then, an event-triggered and compute workflow handles earthquake events. It starts upon reception of the initial earthquake event, when the hypocentre and magnitude are first known, and starts a fast compute path that combines fast tsunami simulations with inundation, generation of a ShakeMap and a fast aggregate loss assessment before the first deadline of 60 seconds after event arrival, deadline suitable for emergency warnings at regional scale for our target scenario, an M8.8 tsunamigenic earthquake west of Padang, Indonesia. Then, upon reception of the earthquake moment tensor which retains more precision than the initial rough data of magnitude and approximate location, a higher precision path is meant to be started, with finer grain tsunami simulations, which would give us an updated ShakeMap and building level loss assessment before a second deadline. Both of those paths require access to the appropriate subset of the exposure dataset to compute the loss assessment, by requesting the right subset from the LEXIS DDI. This allows the jobs to run anywhere on the LEXIS infrastructure and the distributed nature of the DDI ensures that the necessary dataset subset can be retrieved. See Section 4 for the description of both coarse and fine grain TsunAWI simulations.

The output of that compute workflow is used to determine the areas of interest, that is the most affected areas in terms of damage to structures and victims; past events have shown that those affected areas do not always match areas with the highest inundation or highest ground motion acceleration and that it is significant to be able to focus as early as possible on the right areas. Areas of interest are used to plan early tasking of remote sensing products, and simulation results are available to generate the needed geo-referenced products for analysis. The outputs of this final task are conceived to best fit the needs of the satellite-based emergency mapping (SEM) mechanism.

2.2 DETAILED IMPLEMENTATION & IMPROVED DESIGN

A subset of the workflow has been refined and already partly implemented to cope with source uncertainty in the early stage of a tsunamigenic earthquake, and benefit from the performance gains reached on the tsunami simulations. The objective is to build a system able to exploit any amount of available resources and deliver a result in time, demonstrating the following capabilities of the LEXIS infrastructure:

- The ability to run a large amount of short compute tasks (simulations) and to aggregate their result according to pre-defined rules considered suitable for an impact estimate and assessment, while doing it before a preset deadline. This also takes into account the fact that the more results from simulations we do have, the better the accuracy of the resulting aggregation.
- The ability to start the aggregation step as soon as the first simulations are providing results, and so benefiting from the burst buffer accelerators to minimize the time spent waiting for the first results to be available.
- The ability to allocate the tasks on geographically separated computing centres at the same time, to allow for redundancy. Is also included the possibility to run the tasks on Cloud instances instead of a traditional HPC system.

This improved design of the workflow includes: First, a specific task is waiting for an earthquake event. Then, upon reception of that event, this task triggers two parallel setups: a set of tsunami simulations, and a ShakeMap generation plus loss assessment. The tsunami simulations define a subgraph, starting with an ensemble set of source parameters out of the source uncertainty data. This set of sources is used to submit a large number of tsunami simulations, each simulation being short (5 to 8 seconds including I/O, see Figure 2 below). At the same time, a file monitoring daemon is started and the post-processing (aggregation) step is started. The file monitoring daemon sends results as soon as they appear to the post-processing task. When reaching a date close to the time limit (e.g. 2 seconds before the deadline), the post-processing step stops accepting results and produces the final output, before terminating this dataflow subgraph. This subgraph can be launched on both supercomputing centres, at IT4I and LRZ.

As an aspirational goal, in addition to provide redundancy, this architecture can increase the amount of task runs, and so increase the result accuracy. For that, the `Inotify` library from ATOS can theoretically be used to generate the aggregated results. Nonetheless, the first goal will simply be to make it work as long as, at least one of the HPC centre is available. Therefore, in the first step, there will be no requirements of deadline accounted aggregating process between both HPC centres, and the workflow can be run independently from the other.

As another possible merging process, it is also possible to merge the geotiff SEM from earthquake damage assessment and tsunami inundation results from ITHACA processing (see Section 5).

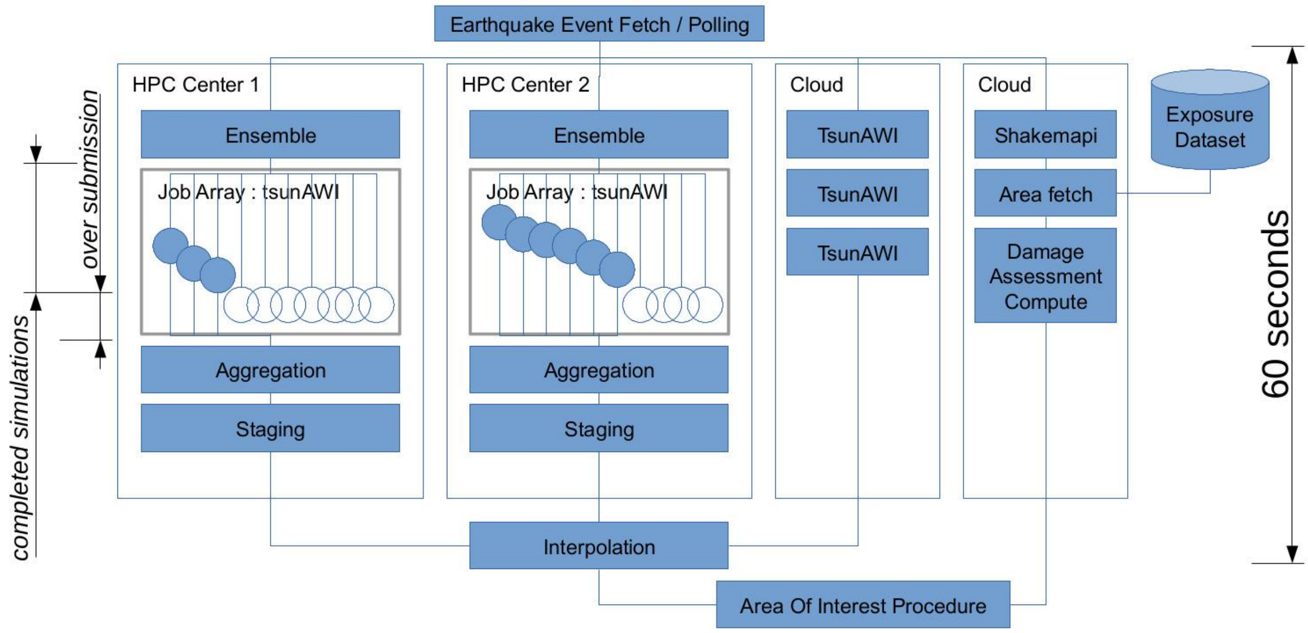

Figure 2 The first stage workflow.
From the first earthquake event information to the area of interest determination, showing how two geographically separated HPC centres and cloud resources are combined.

The complete workflow is detailed in Figure 2 and adds possible other sources of computing using cloud infrastructure (on a best effort base, therefore with less guarantees). This workflow is designed to be robust to compute time and site failure: as long as a single simulation result is available, then we can provide a result to the emergency decision making process.

2.3 TIMING AND ORCHESTRATION

As described in Deliverable D6.2 [4], the implementation of the MoC is done thanks to the implementation of deadlines in the orchestrator. In detail, the orchestration of the tasks and the computations are done as follows.

A detailed view of the tsunami ensemble simulations is shown in Figure 3. Here, we can observe how, in the 60 seconds window (which is still an aspirational goal, but the first goal is —as stated previously— to master the time frame at least within the set of tsunami simulations), three steps are launched simultaneously when the ensemble parameters have been generated. First, the ensemble tsunami simulation jobs, shown on four cores, each job producing a dataset in output. Second, a burst buffer device which stores the produced dataset and emits events for each dataset stored. Third and last, the post-processing step which listens for ready datasets and processes them one by one, before, upon its internal deadline, producing the merged final result. The third step relies on a specific library and system support to monitor datasets being made available and so benefit from the very fast I/O ensured by the second step, and is designed as a monitoring head piped to a post-processing task to decouple the post-processing code from dependencies on the monitoring technology. Thanks to that approach, each tsunami simulation job is encompassed in a script handling the necessary I/O changes for the burst buffer to be used, without having to change the I/O part of the tsunami simulation code.

As for expected execution times for the various operations (Padang Case), we measured or expect the following: ensemble parameters generation is quick (a few hundred milliseconds at most); each tsunami simulation job takes between 5 and 8 seconds and produces between 100 MB and 200 MB of data. And the post-processing of a dataset requires about one hundred milliseconds, with a prevision of two seconds reserved for writing out the collated result. Runtimes of all the jobs are almost constant, except on specific supercomputing nodes and runtime setup where runtimes can vary by a factor of 2 (which is explainable by the node CPU architecture and cannot be controlled).

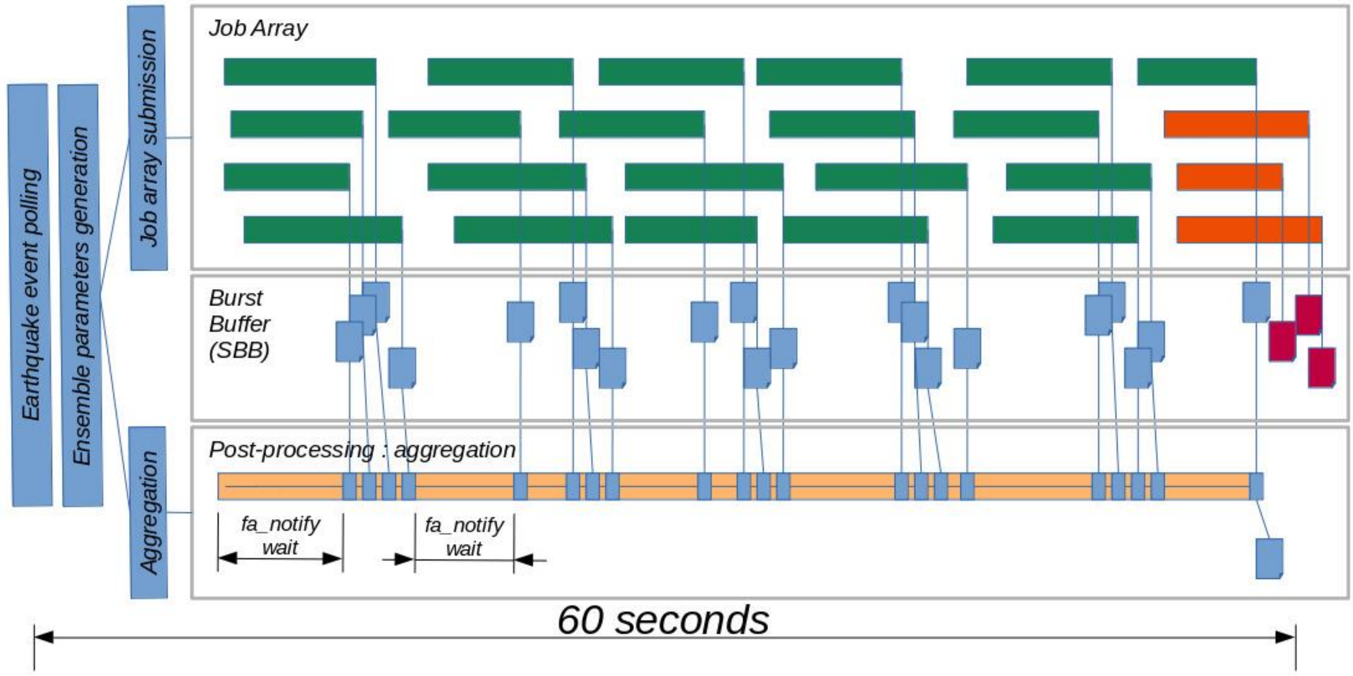

Figure 3 Tsunami ensemble simulation subgraph.
The subgraph details the simulation tasks, with results forwarded through the burst buffer to the post-processing. Red tasks are tasks launched, but whose results are too late to be considered

In this detailed description, we can see how some of the ensemble simulations will be taken into account: some tasks will be planned but will never be launched for lack of time; some tasks will be launched and won't terminate before the deadline and therefore will be killed; and some tasks will terminate, but their results won't be post-processed either because they were written after the post-processing deadline for accepting input, or the post-processing step still has pending results in its input queue at its internal deadline, and those still pending are then discarded. A way to handle these task terminations is described in Goubier et al. 2014 [5].

2.4 USE OF BURST BUFFERS TO SPEED-UP I/O

The burst of activity after an event needs to be processed with real-time constraints within the workflow. Therefore, decreasing I/O as much as possible would help keep jobs within the time window; and a burst buffer can be used to reduce this time.

The burst buffer contains Flash-based, NVME over fabrics memory which can be used to prefetch input data, to provide fast temporary space shared across the workflow steps, and for final results (flushed to global storage after workflow end). Increases in throughput and decreases in latency with respect to traditional storage backends are notable, showing a 600% acceleration with tests using parallel large block writes on NFS.

The burst-buffer space is handled by I/O interception using pre-loaded libraries, so the burst buffer is transparent to the workflow code. Only a different orchestrator configuration must be provided to indicate the directory to be optimized and required storage. Figure 4 provides the schema and data-flow.

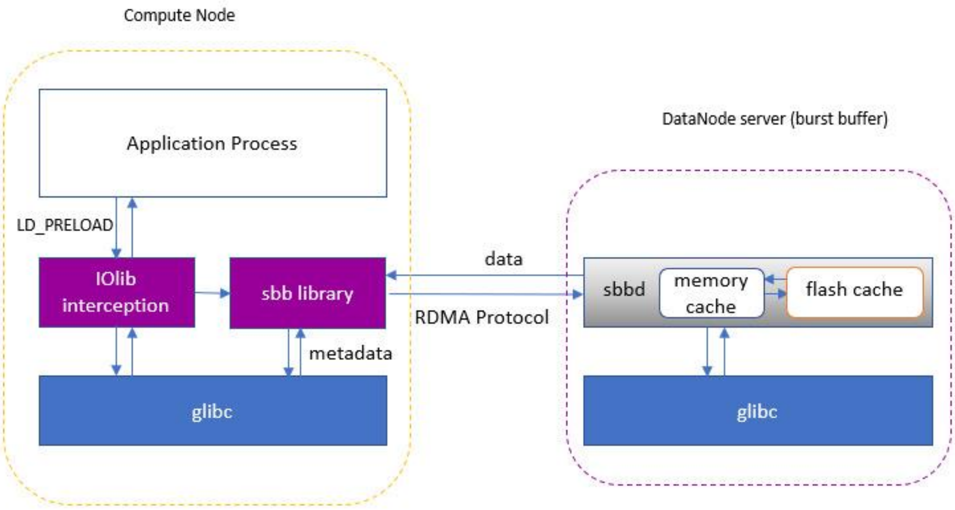

Figure 4 Schema of the flash accelerator system.
Shows how an un-modified application can benefit from the high-speed flash storage at the burst buffer

Using the accelerator means that the supercomputer node can finish the task as soon as the data reaches the burst buffer, so the time limit for node reservations can theoretically be lowered, helping the scheduler place the task earlier, which would help to accept a higher number of tasks in the workflow. The post-processing task is also accelerated by retrieving all previous tasks' results from burst buffer memory.

2.5 ORCHESTRATOR REFINEMENT AND INFRASTRUCTURE

The specific requirements on the orchestration service coming from the urgent computing workflow described in this section are the ability to trigger the workflow when the event (earthquake) occurs and running the multitude of small tasks that would be killed after some period of time.

The first one is solved by a special service listening and waiting for the event data. When these data become available the special script is used to start the first two tasks of the workflows: ShakeMap computation and the TsunAWI simulations. The second problem was solved by creating a special template for YSTIA which starts a job array on the HPC and takes a JSON as an input to set up the parameters for each job in the job array.

The LEXIS DDI Service module is taking care of the distributed data management. The main features are access management on the user and project levels, easy data upload, complex metadata management, distributed staging across several clusters. This means that the user can easily manage data using API, or the LEXIS Portal web interface. Additionally, it complements the orchestration services, which may use data from any available centre and move the data between centres using DDI APIs. Thanks to this the inputs from various data sources of the workflow described in Subsection High-level overview are easily handled and provided for further use almost instantly. In the same way, the results will be immediately available to the emergency services.

Two of the main requirements on the LEXIS DDI were the ability to put the data on multiple centres so the computations may start at as many places as possible, to get as many results as possible, to gather data from different sites and to provide results quickly. The first two requirements can be provided thanks to the staging API which takes care of copying the data from one centre to another and synchronizing the content between the centres. The second part is possible thanks to the DDI API which allows organizations to make direct calls to the LEXIS distributed data interface and use data in their own systems directly.

2.6 SUMMARY AND OUTLOOK

The workflow as implemented by the improved orchestrator demonstrated:

- Resilience with regards to downtime of either the LEXIS cloud infrastructure or the LEXIS HPC infrastructures,
- Guaranteed results after bounded and specified time constraints from the time an earthquake event is detected to the output of preliminary estimations, if at least one of the computing infrastructure is online and reachable,
- Several paths for the workflow allow for more accurate results later, when the fine grained simulations have completed their run.

Ongoing work is still under testing to merge data-elements from the earthquake, the damage assessment and the tsunami evaluation on the whole area of interest/impact, without perceived hard points. For future work, there is still on-going research work on the use of burst-buffers in the context of heterogeneous computing whose first results show promises. Still this work is preliminary and will not conclude within the time-frame of the project. Nonetheless, the use of burst-buffers still can be put into application within the time-frame of the project by accelerating the transfers and merge processes and therefore enabling more tsunami simulations to be run on each HPC centre.

3 OPENBUILDINGMAP AND LOSS ASSESSMENT

3.1 OVERVIEW

Earthquake and tsunami damage and loss assessments require several components to be performed. First, an impact assessment of the event is needed. This describes the spatial distribution of ground motions (ground-motion field) for earthquakes and the spatial distribution of inundation for tsunamis. These impact assessments are then combined with an exposure model describing the affected assets in terms of building types, their reconstruction value and the number of people inside. The connecting piece between impact assessment and exposure data are the vulnerability functions that describe the probability for buildings experiencing different damage states based on the level of impact.

LEXIS is focusing on the optimization of processes that take long to speed up the delivery of damage assessments. Fragility functions are simple and provided by engineers, thus require no optimization. Within LEXIS, we provide the earthquake ground-motion field computation as docker container for easy deployment. Our exposure data is based on OpenStreetMap from which we pull the building data to create the OpenBuildingMap. As part of LEXIS, we have reduced the computational speeds of exposure generation and the data delivery in various ways, described below.

3.2 SHAKEMAP GENERATION

We developed a web engine to generate synthetic ShakeMaps harnessing the OpenQuake engine[1] of the Global Earthquake Model (GEM) foundation. The backend asynchronously digests requests parametrizing earthquake sources in terms of source depth, epicentral location, moment magnitude and focal mechanism. The backend returns shaking in user-definable ground-motion measures (e.g. peak ground acceleration or intensity) and can be retrieved in various formats such as ASCII, GeoJSON, among others. This engine implements an open and documented API that users and other services can query systematically and automatically. An interactive interface (see Figure 5) allows exploring the expected spatial shaking distribution by selecting locations of interest on a map and defining the earthquake source interactively in a web browser. Besides the interactive mode, this service provides, through HTTP

[1] OpenQuake engine: https://www.globalquakemodel.org/openquake

requests, a simple interface for any type of ShakeMap to be used in automated systems that require rapid ShakeMap computations without the need to run local instances of OpenQuake.

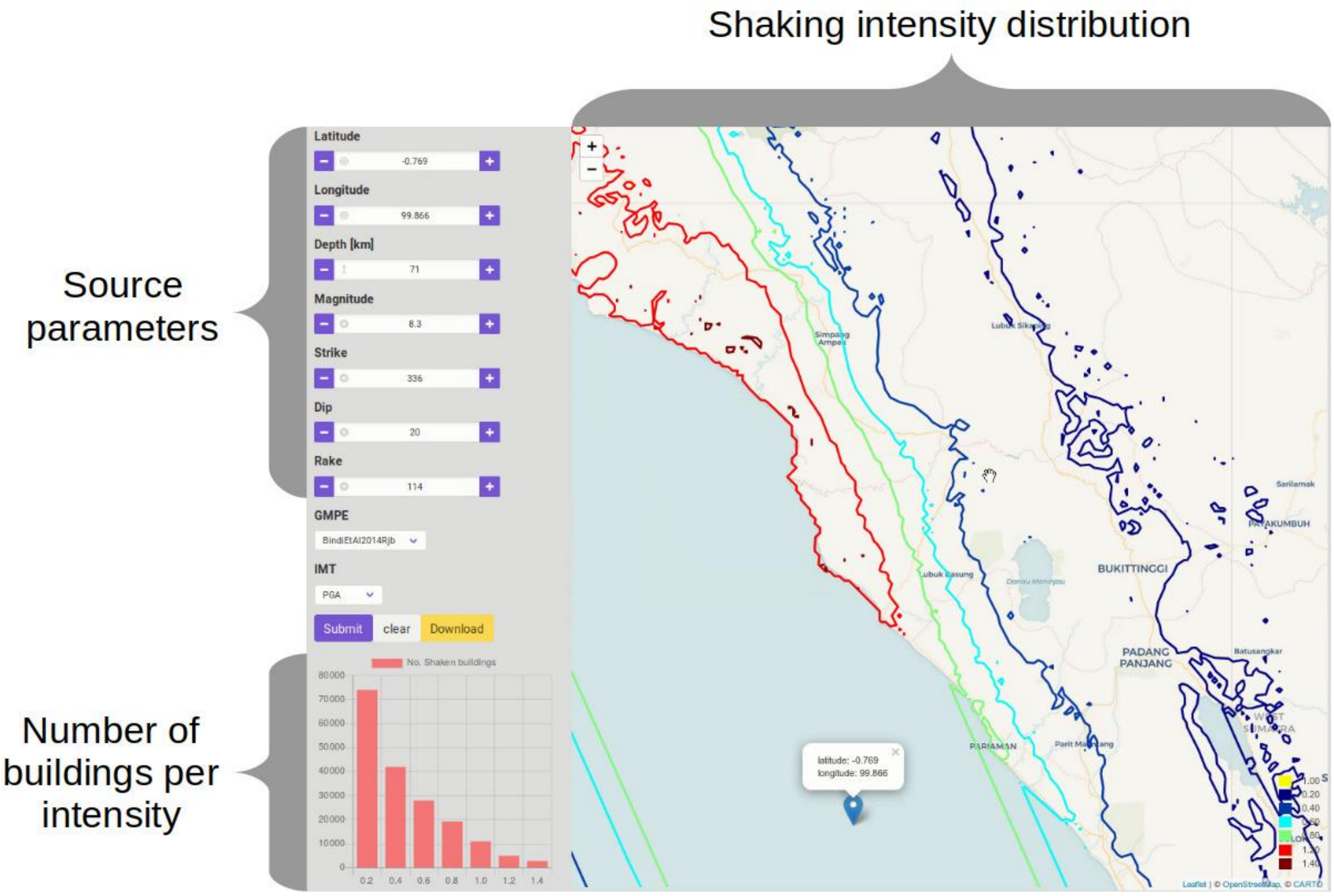

Figure 5 Web frontend of the ShakeMapi.
The user can interactively define the source parameters and select an appropriate ground-motion prediction equation (GMPE). The ground-motion field is then displayed with contour lines on the map and the number of buildings experiencing various levels of intensity are shown as a histogram.

This tool is used to generate the earthquake impact measure for the loss assessment but it also offers various inputs for testing, validating, and benchmarking the effects of ground motion prediction equations (GMPE) on loss assessment scenarios. It is freely available at

`https://git.gfz-potsdam.de/marius/ShakeMapi.`

A repository has been created to host representative scenarios that have been designed and made available to all project partners to ensure efficient streamlining of development as well as compatibility during the development phase. This repository is available to all project partners at

`https://git.gfz-potsdam.de/marius/lexis-scenario-examples.`

The current stable implementation exposes all available ground-motion prediction equations to the user. All of these use the commonly used V_{s30} (the time-averaged shear-wave velocity to 30 m depth) models as site-condition proxy (SCP) to derive the ground motion. As a next step, the V_{s30} SCP will be extended by the slope derived from digital elevation models (DEM). This allows to improve the accuracy of local site amplification predictions. This step will be complemented by a region specific GMPE recommender which will select the best suited GMPE for a given site selection.

A new feature of the frontend is the quantification and visualization of exposed buildings for a given intensity range using building data from the OpenBuildingMap for a rapid assessment of potential damage caused by an earthquake with the given parameters. This feature avoids the more costly computation of a detailed loss assessment for very rapid decision making.

3.3 OPENBUILDINGMAP

The exposure model is purely based on open data to make data distribution easily possible and avoiding problems with conflicting licenses which may lead to the necessity of tailoring data output depending on the consumer. As main open-data source we use OpenStreetMap (OSM) from which we derive the OpenBuildingMap (OBM). OBM can be considered a copy of OSM with focus on buildings in which we extend the building descriptions by explicitly and implicitly given information in OSM or further data sources. OBM estimates all relevant exposure indicators like building type, reconstruction value and people inside in a probabilistic way. These exposure indicators are subsequently used as the main parameters in the loss/damage assessment.

3.3.1 Quadtree

Requesting exposure data on the building scale from a global building database is computationally costly. To ensure fast performance of the OBM in delivering exposure data for rapid loss assessments and to keep the exposure database as small as possible, exposure data is prepared on a geographic tile basis, i.e. all building information of buildings with their centroid in the particular tile are stored as the exposure of that tile.

As reported in D6.2 [4], we transitioned from a regular 30-arcsecond grid, as used frequently in risk modelling, to a more efficiently usable and more flexible grid approach, the Quadtree [6]. The Quadtree data structure approach defines a tile grid across the planet in Web Mercator projection (EPSG:3857). At zoom level 0, there is only one square tile covering the planet (except for the polar regions which in fact are irrelevant for exposure modelling). With each increase of the zoom level z, tiles are subdivided into four squares, hereby increasing the number of tiles, N, to N=4z. At our target zoom level 18, a tile has a width of approx. 150m at the equator and 100m in Europe with the whole grid containing approx. 68 billion tiles due to the cascading nature of the tree. Quadtrees are popular in the mapping community (e.g. slippy map) for delivering raster or vector tiles to map applications, thus allowing our data to be naturally compatible with popular applications like MapSwipe or any web map service. Using an algorithm to aggregate tiles (cascading) into their parent tile allows us to reduce the total number of tiles by a huge margin. Water tiles or empty tiles with no relevant data can be aggregated upwards the hierarchy of the Quadtree as long as all child tiles of the parent tile are empty or contain water. Vice versa, a de-aggregation algorithm allows us to split lower-level tiles into their child tiles.

Using this framework of aggregation and de-aggregation, we calculated a coastline multi-resolution tile grid from OSM coastline vector data. Starting at zoom level 0, we de-aggregated every tile up to zoom level 18 if it contained a coastline, resulting in a set of largest possible tile sizes for tiles covering water or empty land only, while the coastline is described in high-resolution with zoom-level 18 tiles. This de-aggregation reduces the total number of tiles in our database from approx. 68 billion tiles for a zoom-level 18 only grid to approx. 100 million tiles, see the two globe insets in Figure 6. It also allowed us to mask all tiles containing only water. Given that the Earth‘s surface consists of approx. 71% of water, a significant number of tiles have been eliminated so that they are not considered anymore in any type of exposure computation.

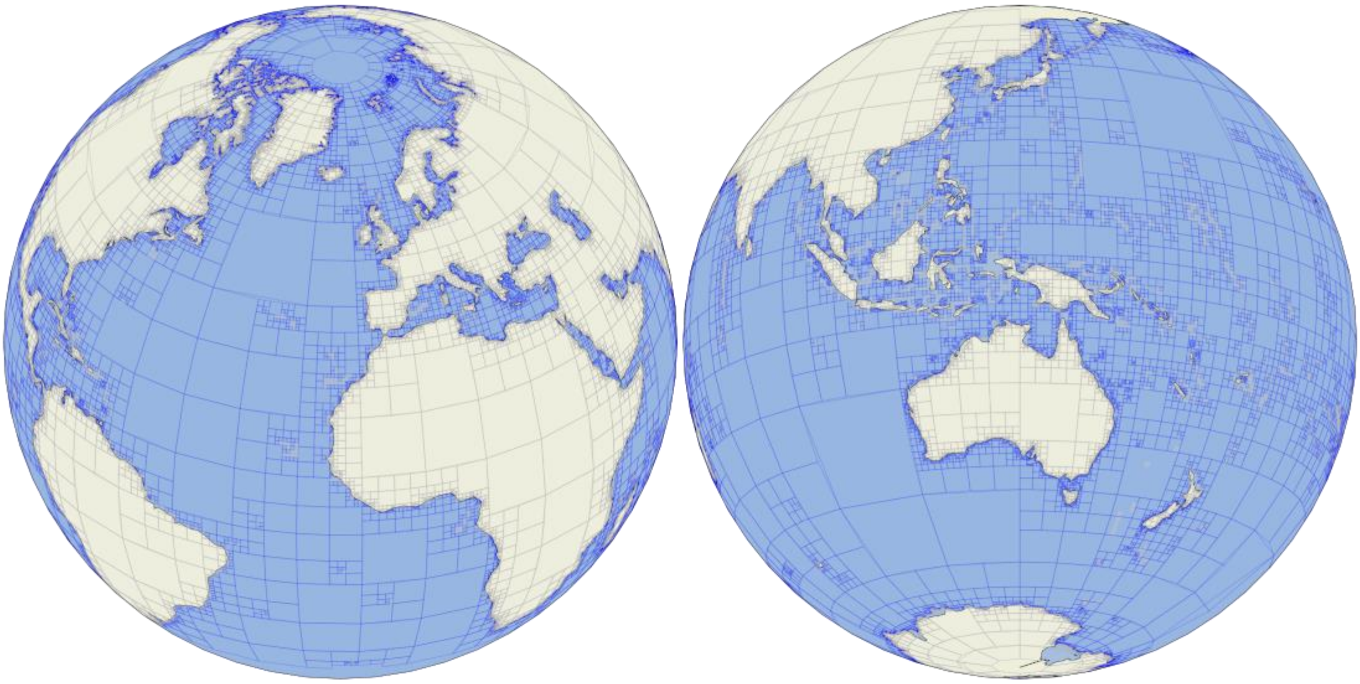

Figure 6 Quadtree structure for the global coastline dataset.
All tiles that contain either land only or water only are aggregated to the maximum size, reducing the total number of tiles from 68 billion to approx. 100 million. Data copyright by OpenStreetMap and contributors.

The simple algorithm for calculating a Quadtree that approaches the coastline is to recursively and top-down quarter the zoom level 1 tiles until a recursion step does not intersect with land or until the target zoom level (here zoom level 18) is reached. Since this requires the knowledge of the real-world coastline, we take advantage of this knowledge and calculate the Quadtree coastline bottom-up starting from the vectorized coastline that is available from OSM. This algorithm operates within the extents of a lower-level tile, e.g. level 9, connects a thin buffer of zoom-level 18 water tiles to the sea-side of the coastline and calculates the remainder of the Quadtree structure in that lower-level tile aggregating tiles where the aggregation is possible. Because the calculation is only bound to a parent tile in which it takes place, the algorithm can be easily parallelized, e.g. by evenly dividing the world into a grid of lower-level tiles of which those that intersect with the coastline are put into the queue.

Calculating the Quadtree coastline for inland water on top of sea water requires to additionally fetch the available inland-water body polygons from OSM and to add them to the source vector datasets. During calculation for classification of water tiles, one only has to distinguish between sea water and inland water tiles. The computation of the Quadtree structure including inland-water polygons is underway.

Furthermore, with mapping applications in mind, we are able to deliver tile geometries from the database or via on-the-fly calculation. Additionally, each coastal tile data entry contains the respective land and water polygons for that tile to preserve that specific information (for non-coastal tiles the tile itself is the polygon).

The next steps will include handling the payload per tile in the tile aggregation/de-aggregation processes. The payload per tile are the aggregated exposure data, the built-area polygons from remote sensing, and the exposure data per building within the respective tile. The current concept foresees the payload to be stored in a directory tree resembling the Quadtree structure and a database for further performance analysis. Through this mechanism, delivery of precomputed exposure data per tile or polygon of interest should be by far faster than accessing the OBM database through expensive geographic queries.

Additional use of the Quadtree structure and the LEXIS DDI data layer is made to allow for an optimisation when a sub-area of the global dataset is needed. A specific routine is able to, from a set of level-18 Quadtree tiles, regroup them in separate sub-datasets by regrouping sibling tiles according to the quadtree parent child structure, until the grouping size has reached a certain threshold. This is because the LEXIS

DDI analysis and optimisation has shown that datasets must be of around 100MB to reach maximum throughput when fetching them: below that threshold, throughput decreases because of the overhead of fetching the dataset. And that fetching a global dataset of a few tens of GB up to a few TB just to work on a small subset of it is wasteful in terms of time, space and energy.

The LEXIS DDI API for datasets took into account that requirement and specifically implemented subset dataset fetching in a simple way, through paths. Once a target geographic area is known, for example from the ShakeMap, then fetching the relevant subset from a global dataset consist of fetching the paths corresponding to the Quadtree tiles covering the target geographic area, following a very similar approach to the way background tiles are fetched when using OpenStreetMap on the web, with the slippy map concept.

3.3.2 Tile completeness estimates

Although OSM contains close to half a billion footprints as of now, building coverage is not complete everywhere. As a consequence, exposure data is delivered in various forms:

1. On a building-by-building level if the area of interest has complete building coverage and if privacy laws allow the distribution (e.g. to civil defence authorities).
2. On an aggregated level per tile if the building coverage is incomplete or if privacy laws do not permit dissemination on the building level.
3. On a mixed level for tiles with incomplete building coverage if privacy laws permit the detailed dissemination. On this level, we provide building information for the buildings in the database and aggregated information for the buildings missing in the database. While this level certainly constitutes a compromise, the whole system gradually converges to the building-by-building level with increasing completeness of the building data in OSM.

We have transitioned the completeness assessment from the previous 30-arcsecond grid to the Quadtree. All completeness estimates are done on level-18 Quadtree tiles as they are the target size for exposure modelling. A tile is considered complete when all real-world buildings are mapped in OSM determined by comparison between satellite imagery and OSM buildings (see Figure 7); likewise, it is considered incomplete when real-world buildings are missing in the mapped data. Additional classes cover unknown for tiles that have not yet been assessed, undecidable when no ground truth can be determined (e.g. due to cloud coverage in the satellite imagery), water or empty when no buildings are present. These four cases are subject to aggregation of quadruplets of tiles of the same completeness assessment, again allowing for reduced requirements in data storage space.

Figure 7 Manual completeness assessment of OBM buildings in the Clickpleteness.
The background shows satellite imagery and is overlayed with the building polygons from OBM (blue hatching). The user can manually select the completeness status (see legend on the bottom right) for each tile (displayed as a grid). Data and map copyright by OpenStreetMap and contributors.

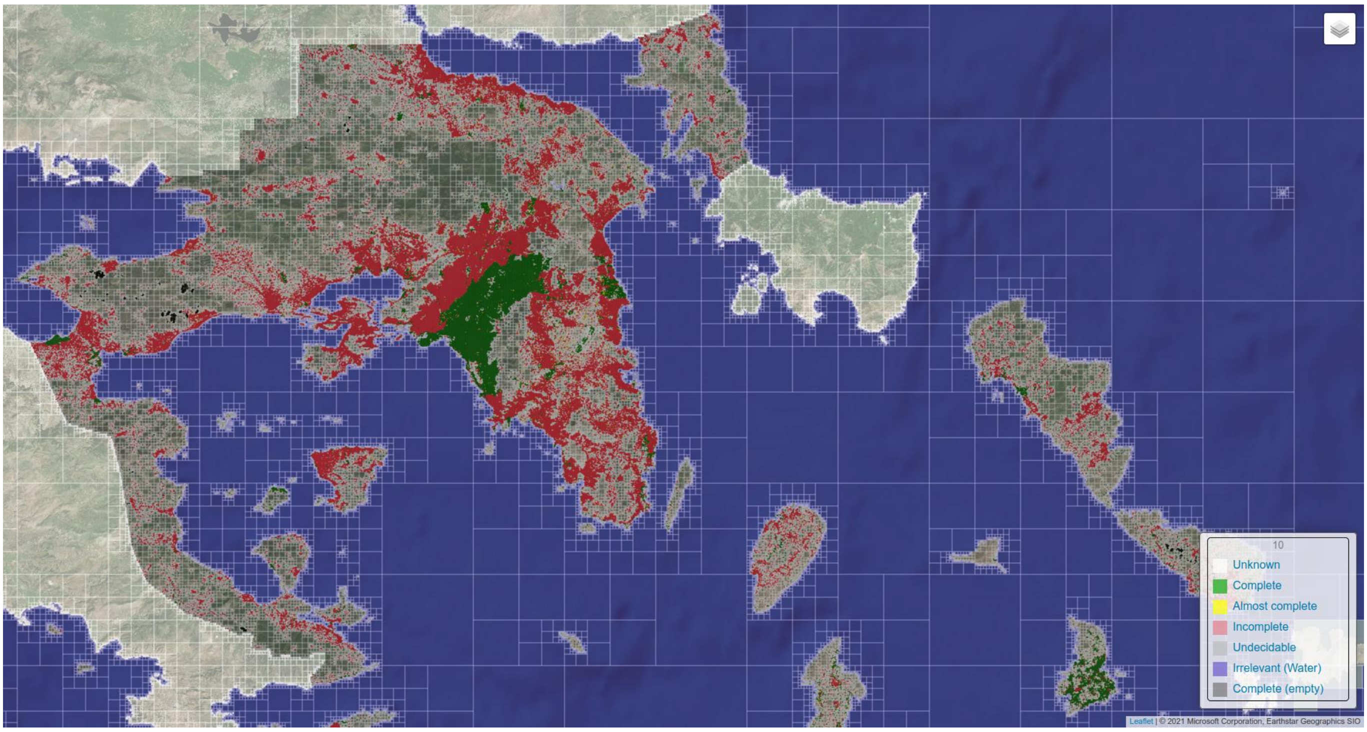

Figure 8 Completeness estimates for the Attica region, Greece.
Green and red tiles indicate complete and incomplete tiles, respectively. Blue tiles indicate water, grey tiles empty land (no buildings) and light grey tiles unassessed tiles. Data and map copyright by OpenStreetMap and contributors. Tiles covering empty land and water are aggregated to the largest possible tile size to save storage and processing times.

As described above, the form of dissemination of exposure data as well as the resolution on which cells are stored depends on their completeness in terms of building coverage. To create these completeness data, we are following a multi-tier approach:

1. We built a website, called *Clickpleteness*, on which an analyst can set the completeness status for one or more cells by comparing the satellite imagery with the overlay of building polygons directly, see bottom-left inset in Figure 7. The *Clickpleteness* enables us to manually modify the underlying Quadtree completeness dataset and to track changes in the OSM building stock data. The building polygon layer (blue hatching) is the current OSM data, and the satellite imagery represents the real-world situation. A tile is considered complete (green) when all real-world buildings are mapped in OSM, else it is incomplete (red). Almost complete (yellow) are cells that can be considered complete for the sake of exposure modelling, albeit missing one or a few buildings. See a large-scale completeness estimate in Figure 8.
2. In a collaboration with the team at the University Heidelberg/HeiGiT, we included a completeness analysis into the MapSwipe smartphone application [7]. MapSwipe has been developed together with the Humanitarian OpenStreetMap Team and is crowd-sourcing the detection of buildings from satellite imagery. The users are presented satellite imagery at the zoom level 18 and they have to decide whether or not an imagery tile contains buildings. This selection is later used to provide people tracing buildings from satellite imagery with a preselection of areas where buildings are to be found. MapSwipe has been extended to be able to now overlay vector data (in our case building footprints) and have the user confirm whether or not the footprints match the satellite imagery, delivering a completeness estimate per tile/cell.
3. Because MapSwipe results so far are publicly available, we can use this data to identify two types of cells: empty cells and incomplete cells. Empty cells (no building at all or water) are the cells not marked by the users as containing buildings at all. Overlaying them with the global coastline, we can distinguish empty cells from water cells. Cells can be interpreted as incomplete if they are marked by the users as containing buildings but no building footprint is present. This approach will not necessarily deliver a completeness classification for each cell as they are undefined states but it will help us to populate our database with completeness information.
4. The last approach is less accurate but can be employed automatically and globally. Instead of manually comparing the OBM building data with satellite imagery, we developed a tool named *OBMGapAnalysis.* It is a processing chain for settlement layers designed to evaluate the completeness state of OBM building coverage on level-18 Quadtree tiles. The input data for this tool is any set of raster files that contain the spatial distribution of human settlements, such as the Global Human Settlement (GHS) built-up area [8] (30 m grid) from the Joint Research Centre (JRC) of the European Commission, the World Settlement Footprint [9] (10m resolution) from the Deutsche Zentrum für Luft- und Raumfahrt (DLR) or any self-produced land cover classification from remote sensing approaches. The processing of these raster files results in gridded values of built-up areas (Figure 9 top-left) that are directly compared with the built-up areas in OSM as defined by the building footprints (Figure 9 top-right). By comparing these values, a completeness state can be assigned (Figure 9 bottom-right). Since there is a considerable difference between the spatial resolutions of OBM and the settlement layers, the tool can optionally take as input the streets from OSM (Figure 9 bottom-left) to refine the built-up model under the assumption that streets are not buildings but are often included in raster files as built-up area. The results of this processing chain are stored in a database for public access and may serve as an input for further analyses.

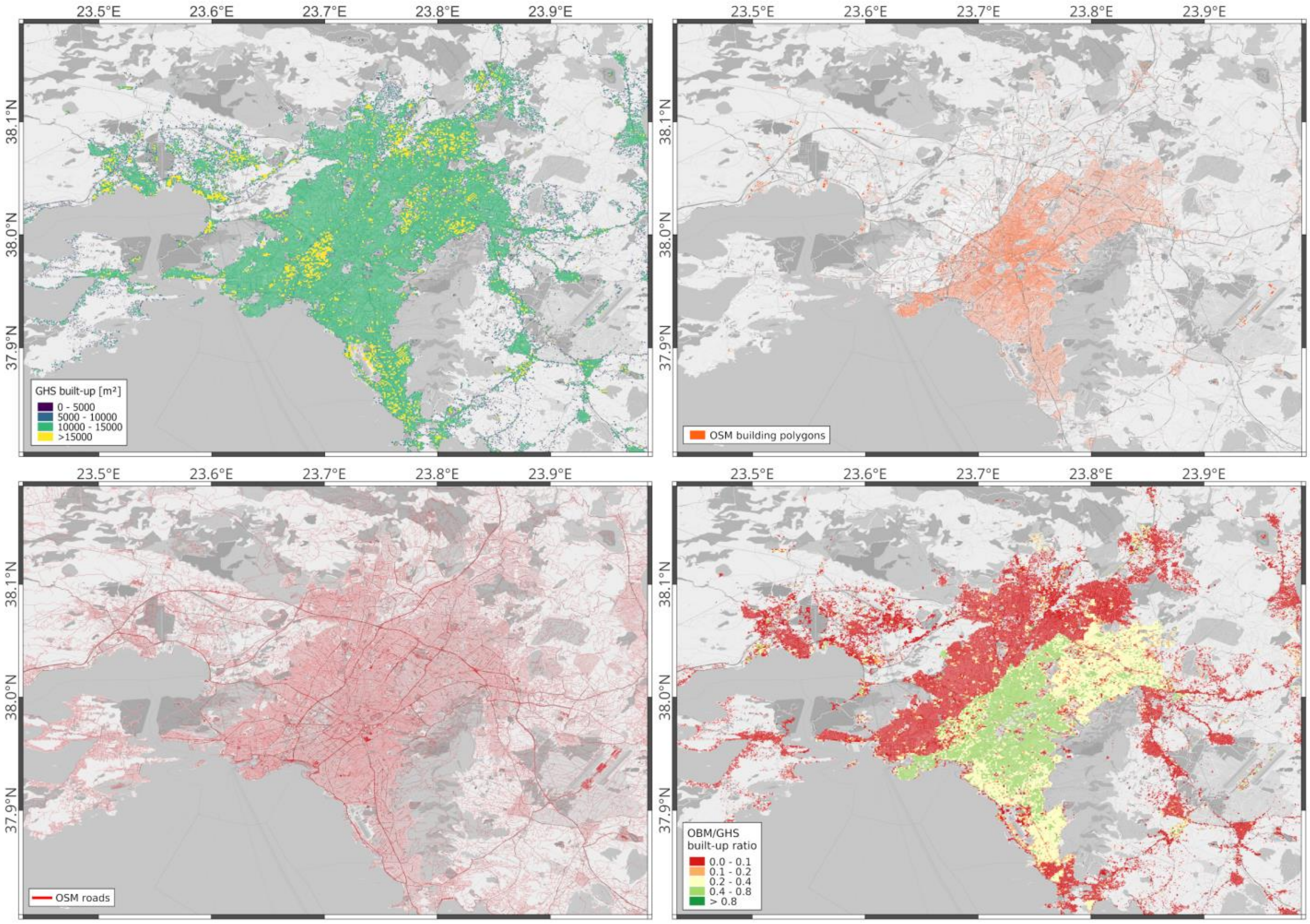

Figure 9 Automated completeness analysis for OBM buildings based on the OBMGapAnalysis tool. The tool imports based on level-18 Quadtree tiles the raster data from the Global Human Settlement Layer (top left). It cuts out all roads as imported from OSM (bottom left). The settlement coverage is then compared to building data in OBM (top right) and the built-up ratio is computed (bottom right). Data and map copyright by OpenStreetMap and contributors.

3.4 EXAMPLE DAMAGE ASSESSMENT

We developed a residential exposure model for the city of Coquimbo, Chile, based on the aggregated exposure data from the SARA project [10]. This model describes aggregates of building types per location. To distribute these aggregates to the buildings from OBM/OSM, we defined Voronoi cells (see inset in Figure 10 and associated every real building to its nearest SARA data location in order to compute the building type distribution from the data given by the SARA model. The preliminary model treats each building as residential for simplicity but will be expanded to cover the other occupancy cases.

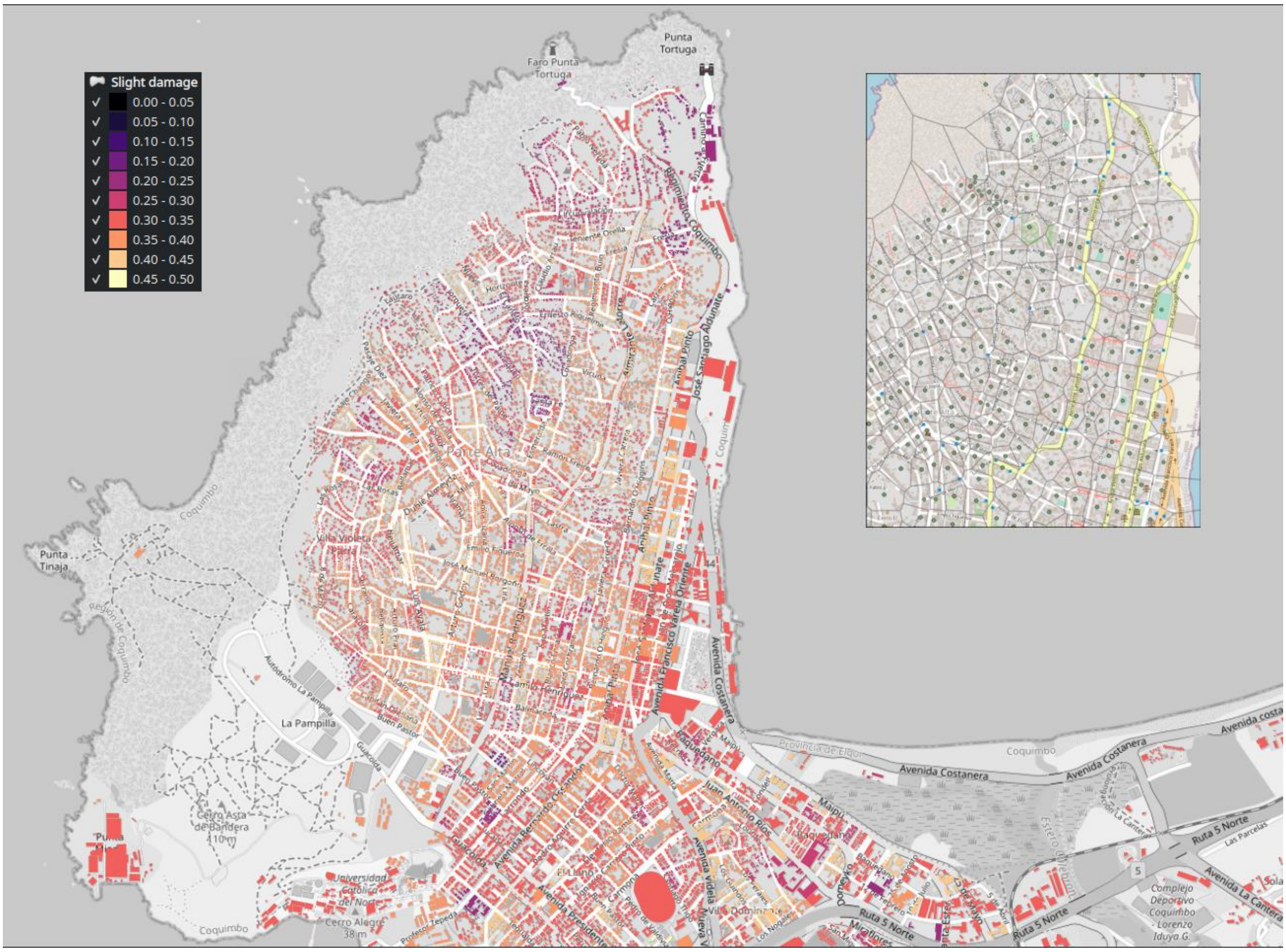

Figure 10 Distribution of probabilities (colour coded) for buildings to experience slight damage due to shaking as caused by the 2015 M8.3 Coquimbo earthquake.
(Inset) Map of the northern part of Coquimbo, with the data points of the SARA model (green circles) and their respective Voronoi cells (grey polygons). Data and map copyright by OpenStreetMap and contributors.

We developed a damage calculator to work with our exposure model. Because this model exposure is described, where possible, at the building level, classical loss calculation needs to be expanded to handle the different types of assets: buildings only, aggregated assets per tile, and a combination of both. Furthermore, new visualizations are needed for the building-level damage assessment.

The calculator interpolates the ground-motion value for all asset locations (either the centre of each tile or a single building). From the taxonomy of each asset, the calculator retrieves the corresponding fragility function and computes the probabilities for each possible damage state. These damage states can be visualized directly at the building resolution (see Figure 10) because all building geometries are processed together with the asset data. However, to enable the user to create classical grid-based damage maps, we compute the aggregated damage per tile from all assets.

Classical exposure models are often provided as rather large CSV files with a lot of repeating information. We have designed a preprocessor to import the aggregated (classical) exposure models into a geospatial database to gain fast access to asset data and their geometries. To remain compatible with classical processing of damage data, we also implemented an export function to provide OpenQuake-compatible CSV files. The next step will be to make the damage calculator operate on a tile-by-tile basis (independent of the resolution of the exposure assets) in order to allow for parallelization of the computation using the newly developed processing engine Rabotnik (see 3.5).

3.5 NEW PROCESSING ENGINE

Generating an exposure model that contains almost half a billion buildings and their related exposure indicators and is based on different data sources such as the specially pre-processed OSM data, satellite imagery, cadastral information as well as our own aggregated exposure models requires a scalable and modular processing engine. To tackle the challenges of Big Data computing in an online operating production system with continuously changing heterogeneous data sources, we designed Rabotnik. It is a Python-based framework for distributed, real-time processing of large datasets, which makes it easy for scientists to define their own processing rules. It was motivated by the exposure model development where several different programmers work on their codes in their expertise but whose results all contribute to the same product.

The engine is built around atomic computational units which we call rules form the entry point for users of the Rabotnik framework. Each rule contains the logic and analysis a programmer wants to execute when one datum changed. For example, the number of stories of a building contained in the OBM (one of many data sources) is updated. The change is detected and will trigger a signal that is being broadcasted into the Rabotnik system. Rabotnik builds on top of existing, solid and state-of-the-art open-source solutions like Celery and RabbitMQ, and uses the Message Queuing Telemetry Transport (MQTT) protocol to broadcast messages and to send data packages between client instances. The emitted signal is sent to a message broker (RabbitMQ) and from there dispatched to clients that are registered to receive the designated signal, see Figure 11.

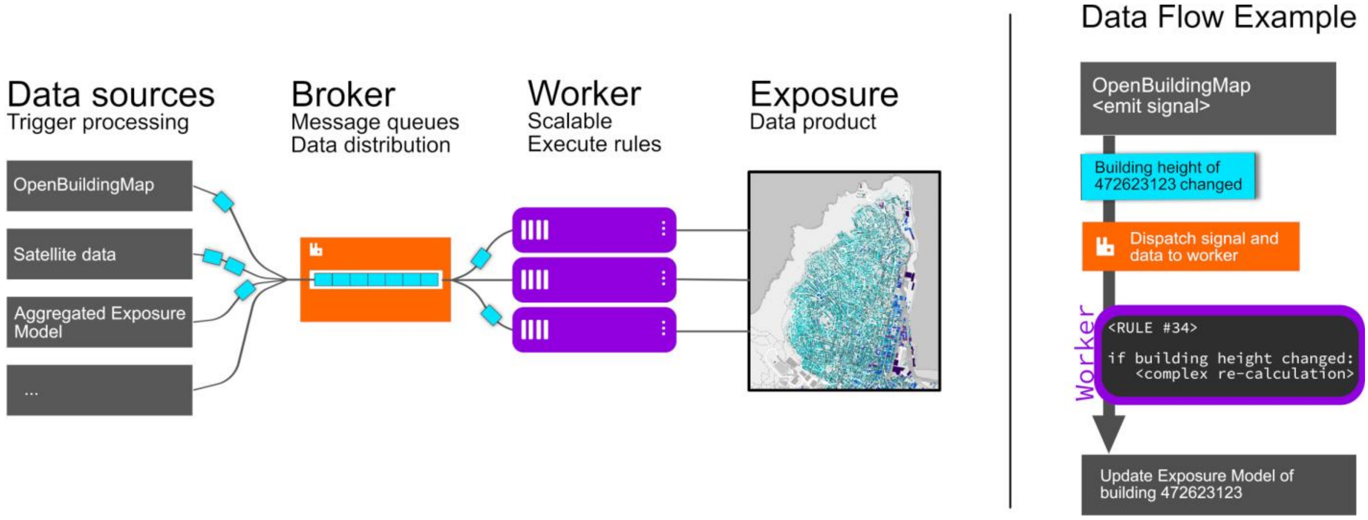

Figure 11 Schematic display of the Rabotnik processing engine.
Dynamically changing data sources push messages to the broker (rabbitMQ) which enqueues the sent data packages before the worker nodes pull and digest them. The rules executed on the workers encapsulate the logic of the exposure model generation and feed their results to the model database. Data and map copyright by OpenStreetMap and contributors.

Each updated, added or removed datum needs to be reflected within the different analyses which all together constitute the exposure model. The highly complex dependencies between individually developed and maintained data-analysis stages require a flexible and extensible task queueing, processing and messaging system that is scalable across computational nodes to enable a real-time processing of the ever changing datasets.

The decentralized architecture allows any participant to subscribe to broadcasting channels and to trigger computations, e.g., if a certain datum such as an attribute of a building within the OpenStreetMap database is updated. The receiving instance can in turn issue a message for other subscribed entities to react upon.

When receiving a signal, a client instance executes the contained rules. Despite versioning the rules, this level of abstraction also allows to track the rules that contributed to generating one datum of the exposure model. This in turn also enables to identify rules that need to be re-executed when an input datum changed. This allows to efficiently identify data that needs to be updated without the need of full re-calculation.

Rabotnik harnesses the Celery framework which allows distributed execution and a high degree of scalability. To avoid blocking code execution in this CPU- and IO-intensive processing, Rabotnik operates fully asynchronously. To further enhance coordination of tasks the message broker organizes the dispatching of messages to clients in message queues. As some tasks are more important than others the queue management allows to set different priorities of message queues. If for example a catastrophic event happens, this allows to focus computation on the area of interest to support a near real-time re-evaluation of the situation.

The current work in progress is published as Free Software, under AGPLv3+ license. The source code is available at

`https://git.gfz-potsdam.de/dynamicexposure/rabotnik/rabotnik.`

4 TSUNAWI

TsunAWI code developments were mostly related to performance improvements with regard to the modelling stage as well as post processing and the quality of data products.

4.1 TSUNAWI PERFORMANCE

4.1.1 Single Precision Arithmetic

As TsunAWI is memory bandwidth bound on multi core architectures, evaluating single precision arithmetic was the last low hanging fruit to pick. At compile time, the parameter WP, "working precision", can be chosen as real64 or real32, and all arrays of floating point numbers are allocated as real (kind=WP). Both simulations of synthetic benchmarks (run up on a sloping beach, Okushiri / Monai beach channel experiment) and realistic setups showed that single precision arithmetic introduces a beneficial filtering of numerical noise without suppressing physical terms.

On a single compute node, the runtime is reduced by 30-50%; see the runtime measurements in the following section on MPI. With MPI and multiple compute nodes, the subdomains become so small that memory bandwidth is not longer a limiting factor. Also, the size of the MPI messages is reduced by a factor of two, however, the latency of the MPI calls remains identical and the effect of the reduced precision decreases.

4.1.2 Parallelization with MPI

Adding MPI parallelization can be regarded as the most important achievement for TsunAWI in the LEXIS project. It overcomes the restriction to one compute node, and the time to solution is no longer determined by the mesh size. Given sufficient compute resources at hand, the dominating limit is now the number of time steps to compute. In particular, real time simulation of the inundation is no longer bound to a regional focus.

We adapt the framework used in the global ocean model FESOM, which shows excellent strong and weak scalability [8]. TsunAWI as a 2D code, however, is not expected to scale as well.

The basis is a domain decomposition with the graph partitioning software METIS [11]. The decomposition only depends on the computational mesh, is performed in a preprocessing step, and is reused for all simulations with the same number of MPI tasks.

Within TsunAWI, three types of MPI communication are employed.

- Exchange of values on ”halo” vertices and elements. Each MPI task holds its own part of the mesh and the adjacent halo vertices and elements. Values on the interface to the neighbouring subdomains have to be exchanged during each time step. Values on the edges are derived from the other halo values and do not need to be communicated. As a start, we employ non-blocking MPI_Isend, MPI_Irecv to overlap the communications with all adjacent subdomains and, if possible, also with independent computations.
- Global collectives are only required for diagnostic purposes, such that non-blocking calls like MPI_I(all)reduce can overlap with a full timestep and do not form a bottleneck.
- The I/O of global fields is still implemented in a serialised way. The master task reads most input data and scatters it to the other tasks. Only the mesh decomposition is stored one file per task and is read in parallel. Finally, the master collects and writes the output. Clearly, the I/O must be tuned, however, as the LEXIS workflow only regards the final output of the maximum inundation depth, output is only required once.

Runtimes for typical TsunAWI scenarios are listed in the following tables. The regional coarse and fine resolution scenarios for the Coquimbo and La Serena areas, Chile, as well as for Padang on Sumatra, Indonesia, were set up for the LEXIS workflow. However, with the MPI parallelisation at hand, we can now also tackle basin wide scenarios in real time. The setup ”Indonesia” was used to compute 10,000 scenarios for the tsunami scenario database at the Indonesia Tsunami Early Warning centre. The computational mesh spans the Indonesian Archipelago from 88° East to 142° East and 22° South to 19° North and resolves all coastal regions with 250m, areas of larger cities with 150m resolution.

Table 1 TsunAWI setups for the Chilean event 2015 and their respective compute time for the time stepping, measured on the IT4I cluster “Karolina” with 2x AMD 7H12, 64 cores in each compute node.

SETUP		COQUIMBO REGION COARSE		COQUIMBO REGION FINE	
NUMBER OF VERTICES		1,709,500		4,888,000	
RESOLUTION		150m-15km		20m-15km	
TIME STEP DT		2.0s		0.15s	
MODEL TIME		10h		10h	
COMPUTE TIME FOR TIME STEPPING WITH SINGLE | DOUBLE PRECISION ARITHMETIC					
1 NODE	128 tasks	60s	137s	3,136s	5,760s
2 NODES	256 tasks	27s	47s	1,410s	2,930s
4 NODES	512 tasks	16s	20s	541s	1,191s
10 NODES	1,280 tasks	11s	12s	267s	269s
20 NODES	2,560 tasks	10s	11s	203s	238s
40 NODES	5,120 tasks	—	—	200s	223s

Table 2 Three different TsunAWI setups for Indonesia and their respective compute time for the time stepping, measured on the IT4I "Karolina"

SETUP		PADANG REGION COARSE	PADANG REGION FINE	INDONESIA
NUMBER OF VERTICES		230,000	1,240,000	11,110,000
RESOLUTION		200m-15km	20m-15km	150m-20km
TIME STEP DT		1.5s	0.15s	1.0
MODEL TIME		2h	2h	24h
COMPUTE TIME FOR TIME STEPPING WITH SINGLE PRECISION ARITHMETIC				
1 NODE	128 tasks	2.7s	116s	3,318s
2 NODES	256 tasks	2s	89s	1,740s
4 NODES	512 tasks	–	55s	766s
10 NODES	1,280 tasks	–	36s	202s
20 NODES	2,560 tasks	–	45s	120s
40 NODES	5,120 tasks	–	–	95s

The runtimes are measured on the new system "Karolina" at IT4I. The strong superlinear scaling for the larger meshes and the very strong effect of single precision is higher than we have observed on older Intel systems (Intel Cascade Lake and Broadwell generation Xeons) and we have to further investigate these effects. Maybe, it is just due to an unfortunate choice of compiler or compiler options for the AMD Rome processor architecture.

TsunAWI can also run in hybrid mode with both MPI as the first level of parallelisation and OpenMP on the task level. First tests gave mixed impressions. Generally, a low number of OpenMP threads, 2-6, was beneficial, in particular to employ hyperthreading. The expectation that mapping each NUMA domain to one MPI task and tackle the load unbalance with OpenMP dynamic scheduling, however, could not generally be met. Also, we did not always observe better scaling by reducing the number of MPI tasks for the same number of compute nodes. Furthermore, hybrid code needs careful handling to map each task and thread to reflect the cluster architecture. Given enough time to prepare a large set of scenarios with one setup, tuning hybrid runs are clearly worth the effort.

The MPI parallelisation is a work in progress, with an initial careful approach, adding both OpenMP and MPI synchronisations generously. More work has to be done. Less used features of TsunAWI have to be added to MPI, e.g., writing raster output and tide gauge data on the fly. Most importantly, the I/O is still serialised and should become asynchronous.

4.2 DATA PRODUCTS

4.2.1 Run-up schemes

The most relevant model output in the LEXIS earthquake and tsunami pilot are the estimates of inundation height caused by the tsunami event. Within TsunAWI there are two alternative algorithms available for the task. One is based on an extrapolation method projecting the major model quantities from the ocean part onto the land part, whereas the second approach rather employs different portions of the governing equations in separated parts of the model domain distinguished by the states wet, dry and right in the transition between the two. The first method allows for higher order calculations by applying numerically cheap or elaborated forms of extrapolations (linear or least-squares) whereas the second method is numerically more efficient and thus better suited for fast estimates. Since TsunAWI operates in triangular meshes, the extrapolation method is additionally complicated by the necessary choice of neighbouring nodes for the calculation of the projections. Since the model TsunAWI was mostly used for the pre-calculation of database scenarios without time restrictions the execution time was a secondary aspect. With the shift towards real time calculations within the LEXIS pilot, optimisations were necessary also in this respect and therefore the two methods were reviewed, evaluated, and further developed. The evaluation considered several standard benchmark experiments, that were carried out for both implementations over a range of parameters for viscosity and bottom friction in order to ensure the consistency of results with respect to energy distribution and general run-up behaviour.

4.2.2 Mesh generation

TsunAWI uses triangular meshes. These allow for flexibility and realistic representation of coastlines and other geographic features, however the design of meshes needs special care since unnecessary high resolution may lead to a very small timestep and consequently unnecessary long execution times. Due to numerical constraints the mesh resolution is coupled to the water depth and thus the mesh quality depends heavily on the available bathymetry and topography data. Since we aim for global coverage, it is necessary to work with freely available data sets and therefore the default base for meshes is the General Bathymetric Chart of the Ocean (GEBCO) augmented by observations of the Shuttle Radar Topography Mission (SRTM) on land. In the recent version GEBCO has a resolution of about 500m whereas SRTM is given at a grid size of about 30m, however with limited vertical accuracy (up to 16m). This data coverage is acceptable for relatively coarse meshes that are used for quick estimates of the inundation estimates. A small section of such a mesh is shown in the figure below. It highlights the smooth transition of the mesh density from the deep ocean to the coastline. More details of this mesh and performance results are included in the table of Section 4.1.2 above (Column: Coquimbo region coarse).

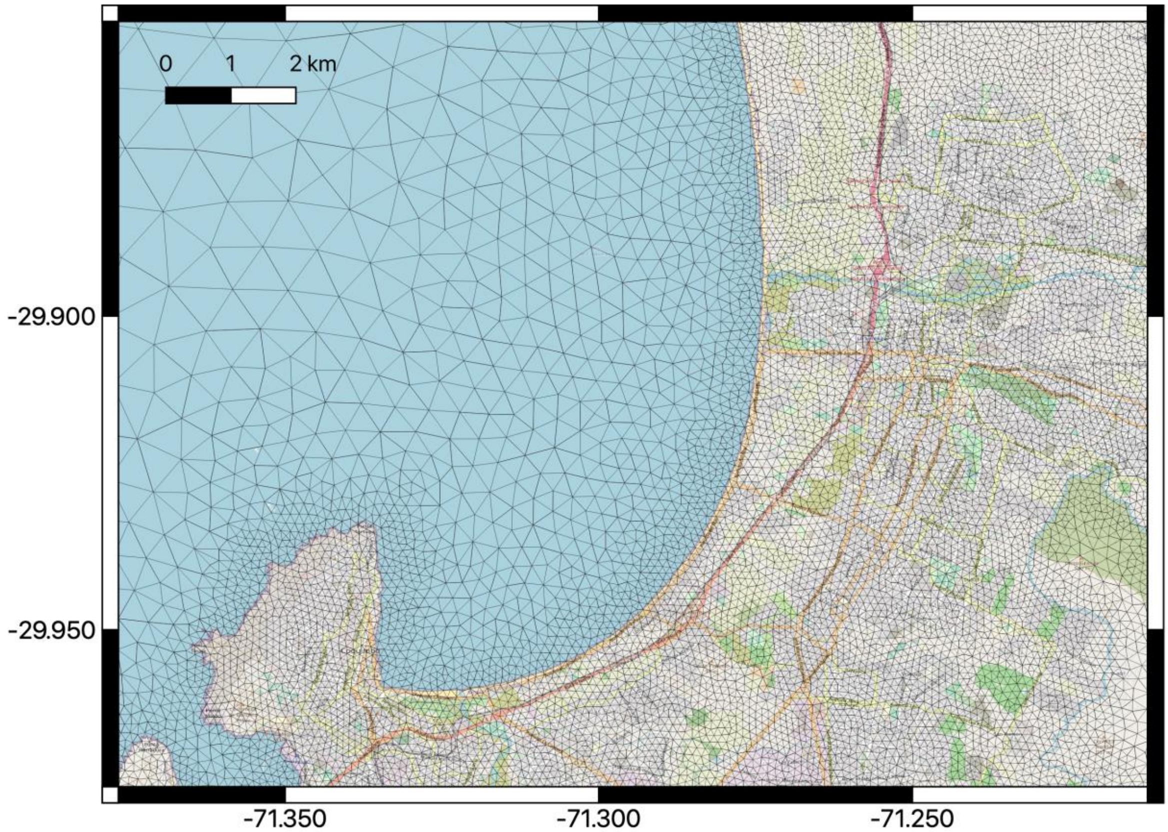

**Figure 12 Small section of a coarse mesh covering part of the Chilean coast.
The mesh is used for fast estimates of the inundation height in the Coquimbo/La Serena region.**

Finer meshes however need adjustments, since coarse data coverage may lead to inappropriate distribution of mesh resolution. An example is shown in the figure below. Again a small section of a mesh with high resolution (approximately 20m) in Coquimbo is shown for two generation strategies. The left panel shows the result of the unmodified algorithm which results in an inappropriate mesh structure close to the coastline due to the discrepancy between desired mesh resolution and considerably lower data coverage. The right panel shows the result of an improved method with better representation of the coast line.

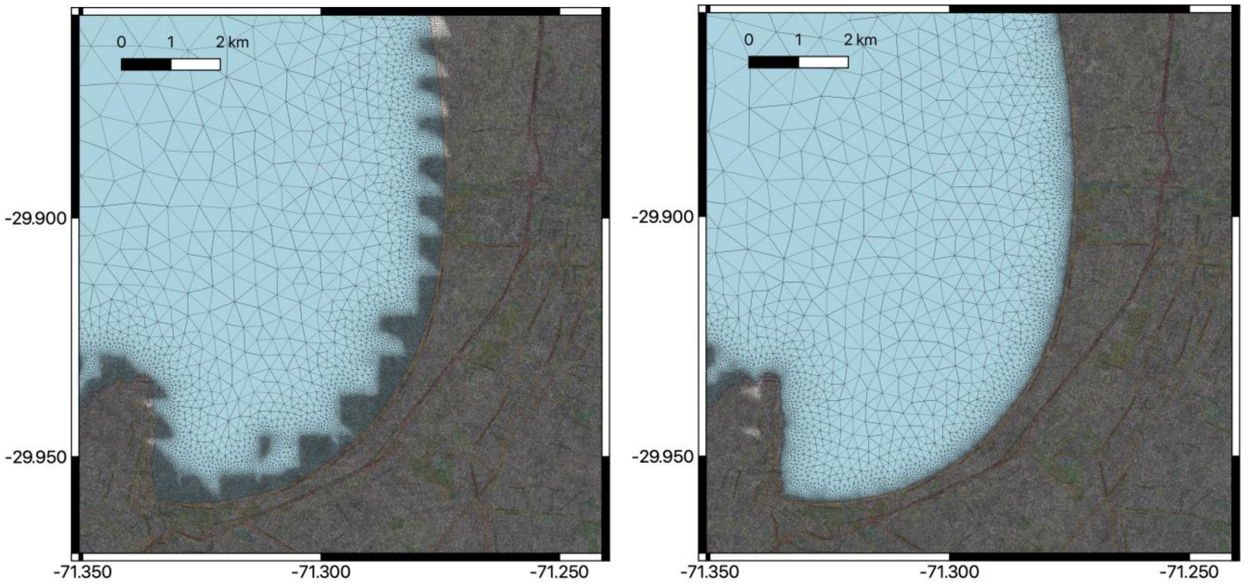

Figure 13 Mesh comparison.
The left panel shows the result of the mesh generation process for a high resolution triangular mesh with low resolution data coverage, whereas the right panel displays the result after better a representation of the coastline. The data remain the same but the resolution in the bay is more appropriate.

The data coverage remains the same, inundation results are not affected considerably, however the mesh density is improved and is more consistent with the actual bathymetric configuration. In summary the mesh generation tools allow for automatic generation of coarse meshes suitable for fast inundation estimates on a global scale, whereas the refined methods with user interaction are used for refined meshes with high resolution in specific tsunami prone locations. Currently, a set of predefined meshes is generated to enable tsunami simulations outside the current focus areas in Chile and Indonesia.

5 IMPACT ON THE INTEGRATED EMERGENCY MAPPING OPERATIONAL WORKFLOW

Satellite-based Emergency Mapping (SEM) system aims to supply information for emergency response by the analysis of satellite imagery.

The Copernicus Emergency Management Service (CEMS) is one of the Copernicus Service and provides data and maps based on satellite imagery before, during and after a crisis in response to a wide variety of disaster events, including earthquake and tsunami.

The CEMS Rapid Mapping (CEMS RM) is a high-speed 24/7 service that covers all the phases of the emergency management cycle, from the satellite tasking and image acquisition until the delivery of maps in support of Civil Protection authorities and humanitarian aid actors dealing with natural and man-made disasters and related emergencies. Since timeliness in data delivery is one of the major requirements in the response phase after a disaster, CEMS service can take great advantages by minimising the time-consuming generation of data using model results from running automated procedures in an HPC environment: the set of procedures described in this section do not require HPC processing per se, but obviously benefit of the computational acceleration in getting the results of the TsunAWI processes run in an HPC environment.

As far as tsunami and earthquake events are concerned, CEMS RM activities would greatly benefit in terms of cost and time for operation by the integration of output derived from simulations integrated with proper exposure information. In particular, the following application and relative improvements in CEMS workflow could be foreseen:

- Supporting for early identification of most impacted areas, to be used for the triggering of satellite acquisitions on which performing damage assessments. Satellite imagery is the main source of data and the early tasking of satellite images, meant as an early satellite acquisition based on predicted timing and potential impact of an event instead of on demand-triggered data acquisition, can significantly improve the delivery time of SEM products [12, 13].
- Enabling the automatic generation of a CEMS RM First Estimate Product (FEP)[2], an early information product that aims at providing an extremely fast (yet rough) assessment of most affected locations within the area of interest. A FEP product based on the results of events' simulations and data on assets exposure, could provide great advantages to the service, both in terms of timelines (procedures are faster than those exploiting satellite data), reproducibility and objectivity.

Based on these considerations, the main goal of this activity is the definition of automatic procedures supporting the fast identification of potentially affected areas derived by earthquake loss assessment dataset and tsunami model outputs with exposed assets as well as the identification, where possible, of the expected damage level. These procedures should support the identification of potential areas of interest to be submitted to the satellite data providers for post-event imagery supply operations, avoiding or reducing any delay. They also enable the production of fully automated FEP products and support the subsequent phases aimed at generating a more accurate damage assessment.

The Chilean earthquake and tsunami events that occurred on the 16th of September 2015, that triggered the activation of CEMS Rapid Mapping[3] (activation ID EMSR137) have been taken as a representative use case to develop the procedure.

5.1.1 Layer in support to AOI definition for an earthquake event

For earthquake events, exposure data about population and buildings, and probabilities of expected damage information at a single building level and provided by loss assessment, have been integrated and subsequently aggregated at a proper map scale suitable for regional level analyses, in order to allow an easy prioritization of areas for the eventual tasking operations.

Specifically, the information provided by a loss assessment and used for the definition of these layers are:

- number of people present in the building in night time hours,
- the probability of damage, subdivided in 5 classes:
    - no damage,
    - slight damage,
    - moderate damage,
    - extensive damage,
    - complete destruction.

All layers produced are based on automatic procedures generating raster outputs using a grid-based approach.

[2] CEMS RM First Estimate Product: https://emergency.copernicus.eu/mapping/ems/rapid-mapping-portfolio
[3] CEMS Rapid Mapping: https://emergency.copernicus.eu/mapping/list-of-components/EMSR137

In detail, three different map products are generated and proposed:

- Relative Potential Damage map, providing the indication of expected most affected areas,
- Damage Absolute Probability map, providing the spatial distribution, in the examined area, of the absolute values of probability of damage derived considering the most severe damage classes,
- Night Population map, providing an estimation of a population exposed to the event in the examined area, based on the population occupancy during night hours.

The Relative Potential Damage (example in Figure 14) product is a normalized index given by the combination of all different classes of damage probabilities and the exposure information about building occupancy. It gives an indication of areas where a higher level of damage is expected, with the aim to help users to identify the spatial distribution of the expected most affected areas. The major limitation of this kind of map is related to its inability to provide an absolute estimation of the expected damage impact level.

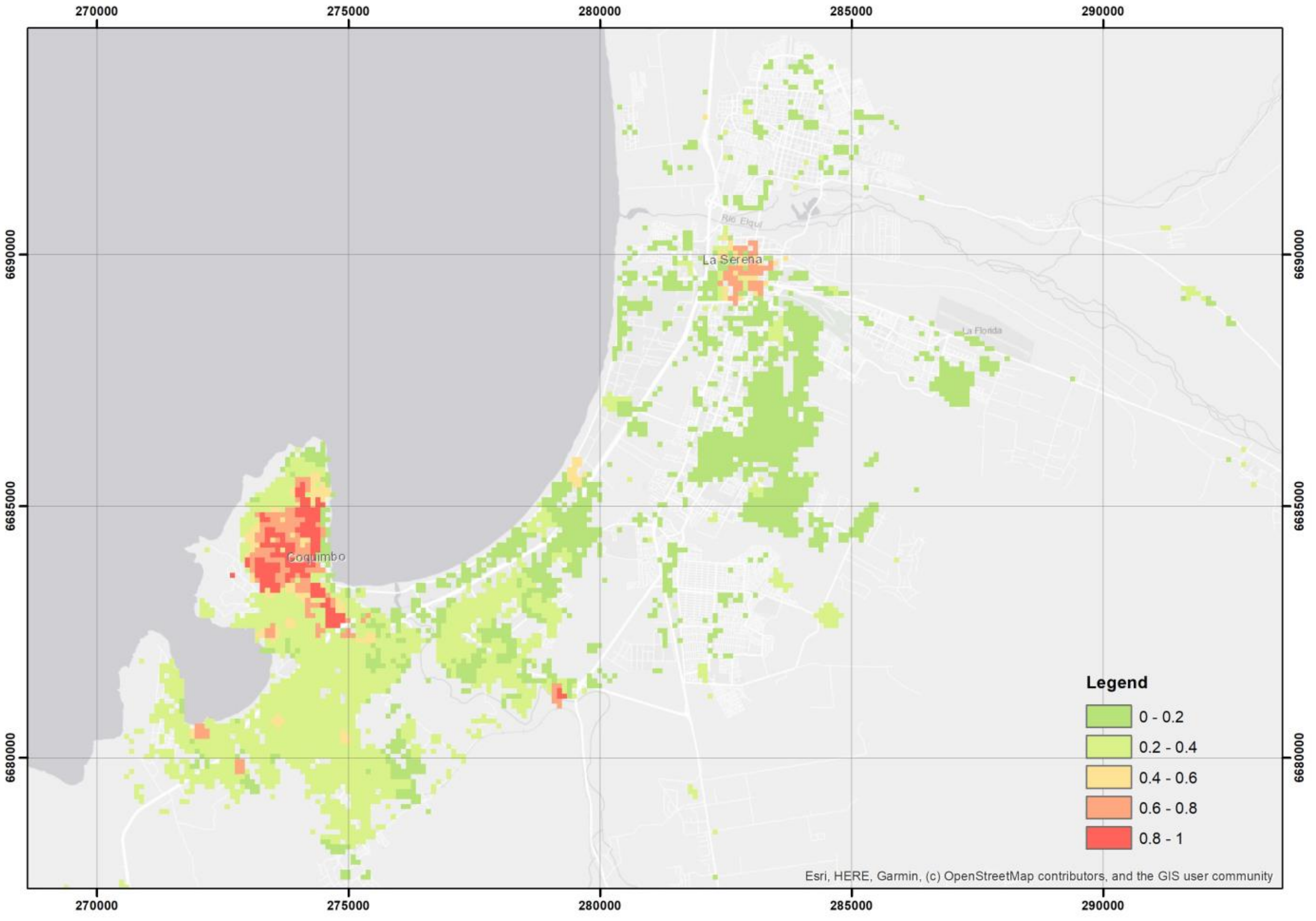

Figure 14 Prioritization of areas of interest based on the Relative Potential Damage map. Coquimbo and La Serena areas

In order to correctly support users in the prioritization of areas for tasking satellite acquisitions, the Damage Absolute Probability product (example in Figure 15) is also proposed to be used in addition to the Relative Potential Damage map and covering different user needs.

The Damage Absolute Probability map provides, for each raster cell, information on the probability to have severe building damages. This product takes into account the maximum values of the sum of the probabilities of having “Extensive” and “Complete” damages, supplied in the earthquakes loss assessment dataset derived from simulations.

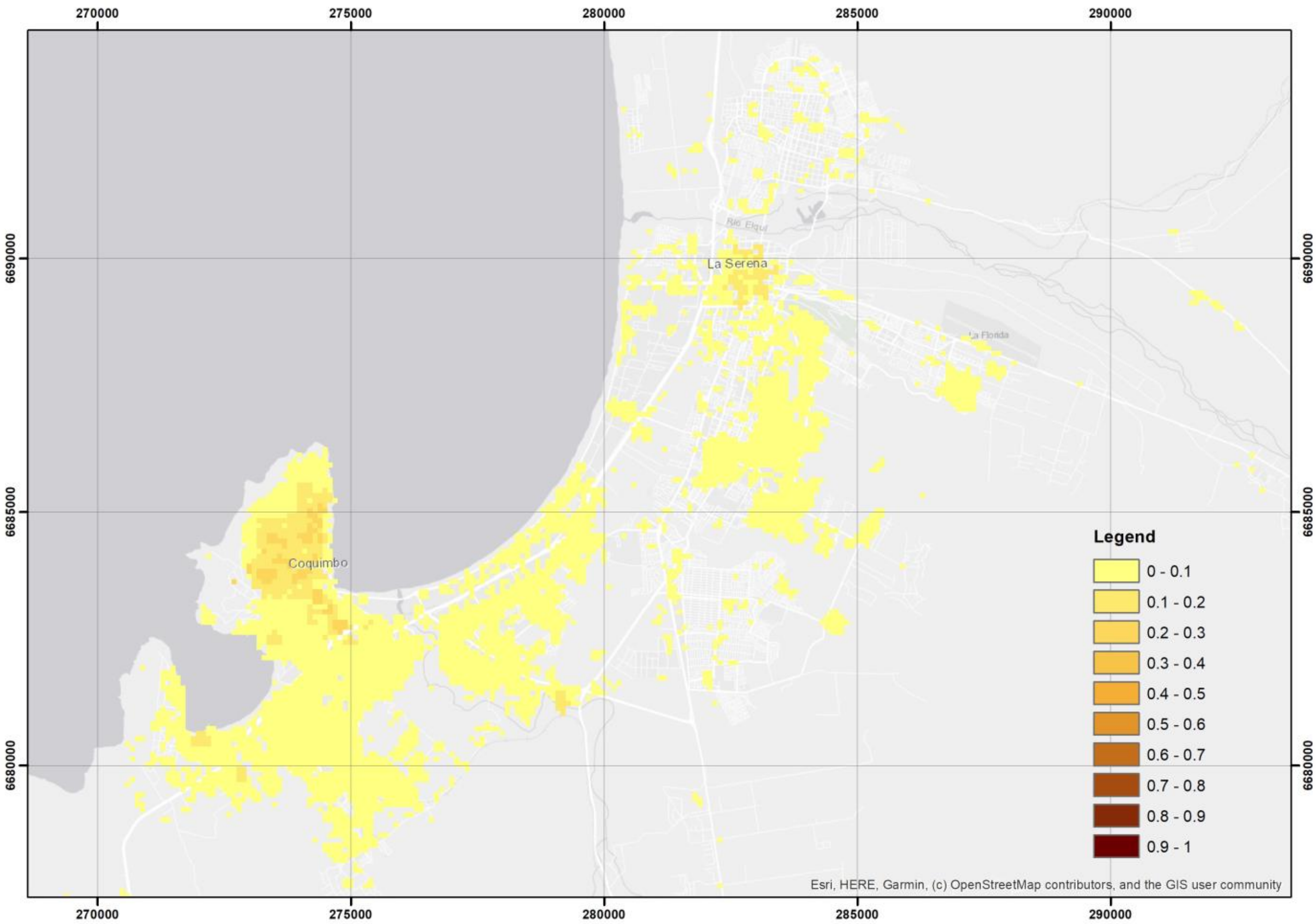

Figure 15 Prioritization of areas of interest based on Damage Absolute Probability map.
In this map, for Coquimbo and La Serena areas a reduced level of criticality is visible, which has been confirmed by satellite-based damage assessment operations carried out in these areas in the frame of the CEMS EMSR137 activation and by information concerning damage to buildings, reported in various field survey results (around 60 houses were destroyed and damaged nearly 200, mostly in the Region of Coquimbo; see, for instance, https://reliefweb.int/node/1177936/)

The main aim of this product is to help users to identify areas that are characterized by higher levels of expected damage to be used for the immediate prioritization of intervention operations. Differently from what was proposed for the previous product, in this approach population exposure information and the probabilities to have less severe damage levels are not taken into consideration.

To produce the Relative Potential Damage and Damage Absolute Probability maps, the maximum value among all buildings included in each raster cell has been taken into consideration, to be conservative and to minimize the risk not to submit to further SEM analysis potentially affected areas.

Night Population map is an additional information layer providing an estimation of a population exposed to the event in the examined area and it's mainly based on the information of night occupancy for each building contained in the cell.

Loss assessment information exploited for the generation of the Relative Potential Damage product, have been used to derive a proper CEMS-RM First Estimate Product (FEP), available in a vector format.

5.1.2 Layer in support to AOI definition for a tsunami event

Concerning tsunami events, the same exposure data about population used in the case of the earthquake event, and the maximum wave height expected over inundated areas generated by TSUNAWI, have been integrated and subsequently aggregated at a proper map scale suitable for regional level analyses, supporting the prioritization of areas for tasking operations and the creation of a FEP.

In detail, information used for the definition of these layers are:

- Building geometry and number of people present in the building during nighttime hours,
- Simulated inundation on coarse mesh (for prioritizing timeliness over accuracy).

As for earthquake events, all layers produced are based on automatic procedures generating raster outputs using a grid-based approach.

As a starting point for all products proposed for the tsunami case, the maximum wave height value has been calculated for all buildings located into the inundated areas, together with exposure data on population occupancy, by an automatic procedure calculating summary statistics of these values for each vector geometry (building footprints) overlapping the raster inundation map.

Furthermore, wave height information collected at a building level has been integrated and aggregated at a regional scale generating different raster outputs. The obtained maps give an indication of areas potentially affected by the event and help also to identify those areas where a higher level of damage could be expected, by means of integrating exposure information about possibly affected buildings, their occupancy and the expected maximum wave height. In particular, two different map products are proposed:

- Potential Damage map (example in Figure 16), providing the indication of areas where a higher level of damage is expected,
- Building Exposure map, providing an estimation of the number of buildings exposed to the event.

Potential Damage is a normalized index given by the combination of wave height values over inundated areas provided by simulations with TsunAWI coarse simulation and the exposure information about the occupancy.

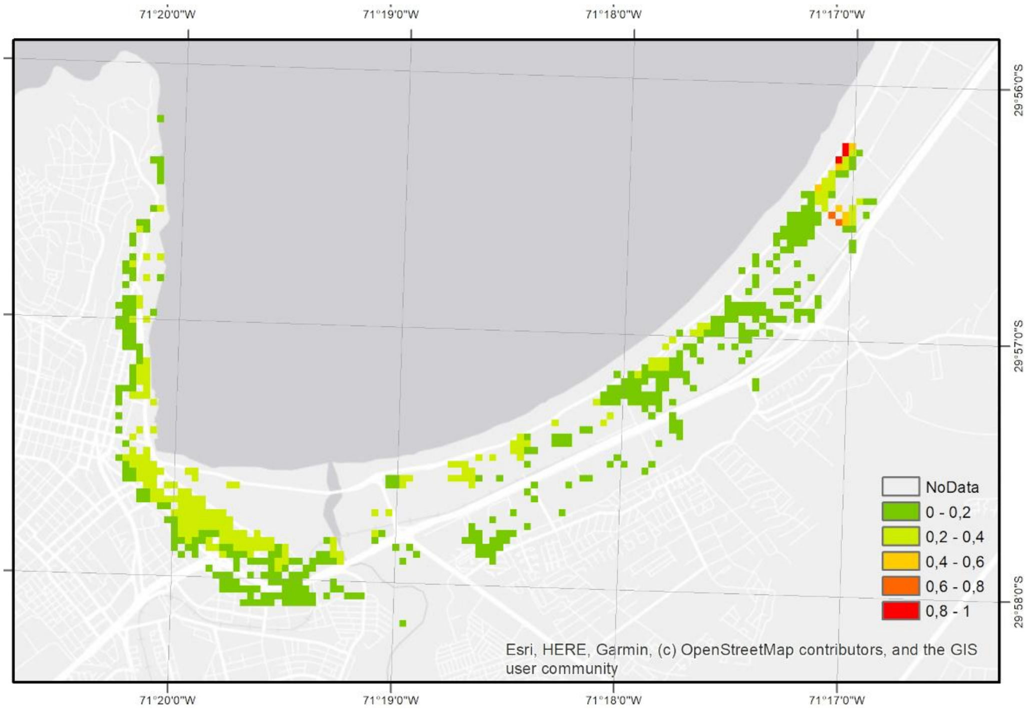

Figure 16 Prioritization of areas of interest based on the Potential Damage map. Detail on the Coquimbo area.

Building exposure map is an additional information layer providing an estimation of a population exposed to tsunami events and it's mainly based on the information of night occupancy for each building contained in the cell.

Obtained vector building features affected by the inundation, including information about people potentially affected and the expected wave height to which they could be subjected, constitute the FEP based on simulated data. As for earthquake event, FEP products can be available in vector format.

6 SUMMARY

This document presents the evolution and work undertaken on the components of the earthquake and tsunami large scale pilot. It charts how the workflow itself has evolved, its scheduling being detailed, and how the components have been improving during the Task 6.3: the model of computation implementation through the use of the orchestrator, OpenBuildingMap, the ShakeMap generation, and the TsunAWI tsunami simulation code, Satellite-based Emergency Mapping (enacting as data fusion), as well as the directions we are focusing on for the remainder of that task.

The most notable improvement was made on the OpenBuildingMap updates and TsunAWI. In case of OpenBuildingMap, a change from regular 30-arcsecond grid to a Quadtree tile representation was made. This was a complete redesign of the system which lead to a much more scalable system. Additionally, Rabotnik system was created to handle asynchronous request for updates and computations. Thanks to this, the system became much more scalable. In case of TsunAWI, several improvements were made, with many optimizations of the code and reordering of the mesh to improve the performance. The benchmark results measured on the Karolina cluster were provided.

In case of Satellite-based Emergency Mapping, the main changes were to enable the usage of the TsunAWI and ShakeMap outputs in the system by automated way and preparation of the scripts which creates a geotiff products containing the information about the tsunami inundation level and building affected by the earthquake.

REFERENCES

[1] T. Goubier, A. Ajmar, C. D'Amico, P. Dubrulle, S. Grita, S. Louise, J. Martinovič, T. Martinovič, N. Rakowsky, P. Savio, D. Schorlemmer, A. Scionti and O. Terzo, “Earthquake and Tsunami workflow leveraging the modern HPC/Cloud environment in the LEXIS project,” in *Proceedings of 22nd International Conference on Network-based Information Systems (NBiS)*, 2019.

[2] V. Silva, H. Crowley, M. Pagani, D. Monelli and R. Pinho, “Development of the OpenQuake engine, the Global Earthquake Model's open-source software for seismic risk assessment,” *Natural Hazards,* vol. 3, no. 72, pp. 1409-1427, 2014.

[3] P. Dubrulle, C. Gaston and N. Kosmatov, “A Data Flow Model with Frequency Arithmetic,” in *Proceedings of International Conference on Fundamental Approaches to Software Engineering (FASE)*, 2019.

[4] LEXIS Deliverable, *D6.2 Pilots Improvements: Solutions Adopted.*

[5] T. Goubier, D. Couroussé and S. Azaiez, “т C: C with Process Network Extensions for Embedded Manycores,” in *ICCS 2014*, 2014.

[6] R. A. Finkel and J. L. Bentley, “Quadtrees a data structure for retrieval on composite keys,” *Acta Informatica,* vol. 4, p. 1–9, 1974.

[7] T. Ullah, S. Lautenbach, B. Herfort and D. Schorlemmer, “(in revision). Assessing completeness of OpenStreetMap building footprints using a gamification approach via MapSwipe”.*Geo-spatial Information Science.*

[8] A. Florczyk, C. Corbane and et al., GHSL Data Package 2019, EUR 29788 EN, Luxembourg: Publications Office of the European Union, 2019.

[9] M. Marconcini, A. Metz-Marconcini, S. Üreyen, D. Palacios-Lopez, W. Hanke and et al., “Outlining where humans live, the World Settlement Footprint 2015,” vol. 7, no. 1, pp. 1-14, 2020.

[10] C. Yepes-Estrada, V. Silva, J. Valcárcel, A. B. Acevedo, N. Tarque, M. A. Hube and et al., “Modeling the Residential Building Inventory in South America for Seismic Risk Assessment,” *Earthquake Spectra,* vol. 1, no. 33, p. 299–322, 2017.

[11] N. Koldunov, V. Aizinger, N. Rakowsky, P. Scholz, D. Sidorenko, S. Danilov and et al., “Scalability and some optimization of the finite-volume sea ice-ocean model, version 2.0 (fesom2),” *Geoscientific Model Development 12,* p. 3991–4012, 2019.

[12] G. Karypis and V. Kumar, “A fast and high quality multilevel scheme for partitioning irregular graphs,” *SIAM Journal on Scientific Computing,* vol. 20, p. 359–392, 1998.

[13] A. Ajmar, A. Annunziato, P. Boccardo, F. G. Tonolo and A. Wania, “Tsunami modelling and satellite-based emergency mapping: Workflow integration opportunitiesart. no. 314.,” *Geosciences,* vol. 9, no. 7, 2019.

[14] A. Wania, I. Joubert-Boitat, F. Dottori, M. Kalas and P. Salamon, “Increasing Timeliness of Satellite-Based Flood Mapping Using Early Warning Systems in the Copernicus Emergency Management Service,” *Remote Sens.,* vol. 13, no. 11, 2021.