Zero-Shot Satellite Image Retrieval through Joint Embeddings: Application to Crisis Response

James Walsh[*] University of Cambridge

William Fawcett[*] University of Cambridge

Grace Colvard[*] University of Cambridge

Raúl Ramos-Pollán[*] Universidad de Antioquia

Abstract

Semantic search of Earth observation archives remains challenging. Visual foundation models such as CLAY produce rich embeddings of satellite imagery but lack the natural-language grounding needed for intuitive query, and full contrastive training of a remote-sensing CLIP-style model requires paired data and compute that are unavailable at global scale. We present GeoQuery, a zero-shot retrieval system that sidesteps this constraint through prompt-aligned text proxies. Rather than training a joint encoder, we generate language descriptions for a 100k proxy subset of global Sentinel-2 tiles and optimise the description-generation prompt so that distances in the resulting text-embedding space correlate with distances in the frozen CLAY visual-embedding space. Queries are resolved in two stages, with a text-similarity search over the proxy subset followed by a visual nearest-neighbour search over worldwide CLAY embeddings. On 76 disaster-location queries covering UK floods, US wildfires, and US droughts, GeoQuery achieves 31.6% accuracy within 50 km, with the strongest performance on floods (50% within 50 km) where terrain features are well captured by RGB embeddings. Deployed within ECHO, a crisis response system using Agentic Action Graphs, GeoQuery identified vulnerable areas during Brisbane's 2025 Cyclone Alfred, with downstream flood simulations reproducing historical patterns. Prompt-aligned proxies offer a practical bridge between EO foundation models and operational retrieval when full contrastive training is out of reach.

1 Introduction

Earth observation (EO) archives have grown faster than the tooling to query them. Operators searching for disaster-relevant imagery, "areas vulnerable to flooding," "recent burn scars," must today fall back on coordinate-based queries or manual inspection, because the strongest visual foundation models for satellite imagery, such as CLAY [1], produce rich embeddings without text grounding. The natural fix, training a remote-sensing CLIP-style model end-to-end, requires paired image-text data and contrastive compute that are out of reach at global scale.

GeoQuery[2] is a zero-shot retrieval system that achieves natural-language access to global Sentinel-2 imagery without end-to-end contrastive training, by aligning text and image spaces *indirectly* through prompt-aligned proxies. We sample a 100k subset of global tiles, generate a language description for each via a vision-language model, and optimise the description-generation prompt so that pairwise distances in the resulting text embeddings track pairwise distances in the frozen CLAY visual embeddings. A natural-language query is then resolved in two stages. The first

[*]Equal contribution.

[2]GeoQuery code: `https://github.com/rramosp/geoquery-poc`

performs a text-similarity search over the 100k proxy descriptions, and the second performs a visual nearest-neighbour search over worldwide CLAY embeddings, using the retrieved proxy tiles as visual anchors.

We demonstrate GeoQuery within ECHO, an operational crisis response system using Agentic Action Graphs (AAGs) to orchestrate complex workflows, where it supported Australia's National Emergency Management Agency during Brisbane's 2025 Cyclone Alfred. Whilst crisis response provides our motivating application, the contribution is the prompt-aligned proxy method itself, together with an empirical map of where indirect alignment suffices and where it does not, such as flood-prone terrain (well captured) and ephemeral wildfire scars in RGB (poorly captured). We situate GeoQuery against three lines of prior work, namely vision-language foundation models, EO-specific representation learning, and LLM-based agentic orchestration.

The remainder of the paper is organised as follows. Section 2 reviews vision-language and EO foundation models alongside prior work on agentic orchestration. Section 3 describes the construction of the joint embedding space and the two-stage retrieval procedure. Section 4 reports the ablation study and the Cyclone Alfred deployment, and section 5 interprets the failure modes and outlines further work. Appendices A and B give the embedding structure and the full ablation tables, appendices C and D document the ECHO crisis-response framework and its tool library, and appendix E presents CrisisSim case studies.

2 Background

Vision-language models have made image retrieval possible through joint embedding spaces. CLIP [2] demonstrated cross-modal alignment at scale by training on 400 M image-text pairs, enabling zero-shot transfer to novel domains, and subsequent work has extended this approach with larger noisy corpora [3], captioning-based bootstrapping [4] and improved contrastive objectives [5]. Recent EO-specific models build on this foundation but face unique challenges. CLAY [1] provides multi-spectral, multi-resolution embeddings through masked autoencoding on Sentinel and Landsat imagery, capturing seasonal and atmospheric variations but lacking text grounding. Prithvi-EO-2.0 [6], trained on 4.2 million global time series samples from NASA's Harmonised Landsat and Sentinel-2 (HLS) archive at 30 m resolution, performs land-cover classification well but requires task-specific fine-tuning. SatMAE [7] employs temporal and spectral masking strategies tailored to satellite imagery's unique characteristics. A wider family of EO foundation models, including Scale-MAE [8], SatlasPretrain [9], SpectralGPT [10] and the multi-sensor DOFA [11], address spatial scale, dataset coverage and spectral richness, with a recent survey provided by [12] and a unified evaluation suite by GEO-Bench [13]. GeoRSCLIP [14] and RemoteCLIP [15] adapt CLIP architectures for remote sensing using the 5 M-pair RS5M corpus and a corpus assembled from heterogeneous box and mask annotations, whilst SkyScript [16] and DOFA-CLIP [17] extend the contrastive paradigm with semantically diverse pairs and multi-sensor inputs, but all rely on paired image-text data that does not exist at a globally consistent scale.

LLM-based orchestration has been applied across scientific domains, building on foundational work in self-taught tool use [18], reasoning-action interleaving [19] and multi-agent conversation frameworks [20]. ChemCrow [21] augments large language models with eighteen chemistry tools to plan and execute syntheses, whilst Coscientist [22] drives end-to-end experimental automation, including the optimisation of palladium-catalysed cross-couplings. ProtAgents [23] coordinates protein design workflows across multiple specialised models, and GeoGPT [24] demonstrates applications to geospatial analysis but lacks the human oversight required for high-stakes scenarios. These systems decompose complex tasks into tool invocations, but crisis response demands additional constraints, such as interpretability, human validation gates, and deterministic verification. For this work, we place GeoQuery inside of ECHO, an AAG processor, more details in appendix C.1.

Earth observation presents distinctive challenges not encountered in natural image domains. The integration of optical, synthetic-aperture radar (SAR) and hyperspectral modalities, each governed by distinct physical principles and noise characteristics, creates considerable heterogeneity. Temporal variations arising from seasonal cycles, phenological processes and atmospheric conditions fundamentally alter scene appearance. Moreover, features manifest differently across spatial scales, from sub-metre commercial imagery to 10 m Sentinel-2 acquisitions, whilst geographic and sensor-specific variations introduce substantial domain shifts. Several benchmarks address specific crisis-relevant

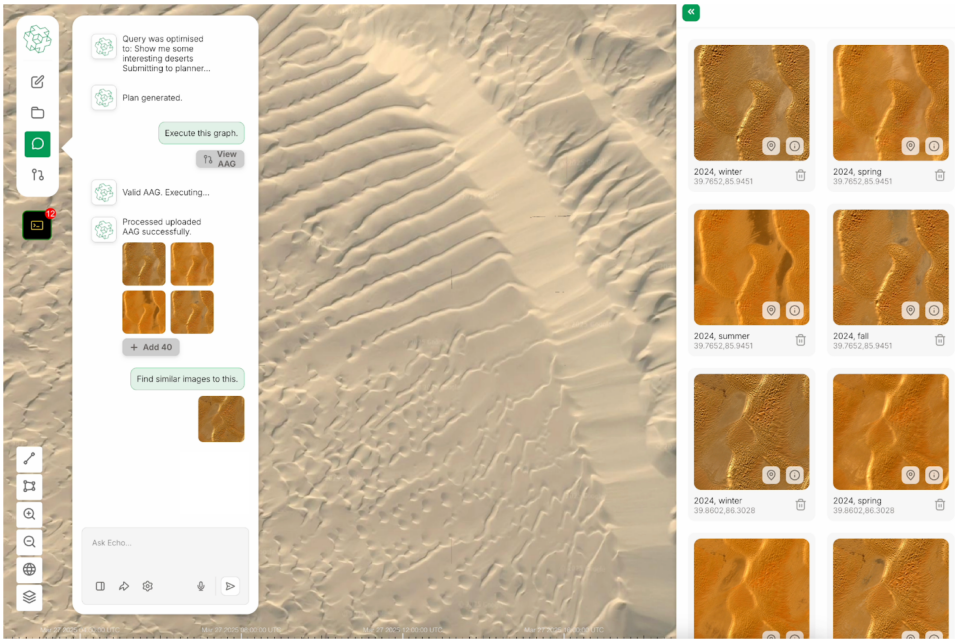

Figure 1: The GeoQuery interface within ECHO, showing the natural-language navigation ("show me deserts") and the similarity search.

tasks at scale, such as flood segmentation in Sen1Floods11 [25], global flood mapping in WorldFloods [26] and post-disaster building damage assessment in xBD [27], but each requires task-specific labels rather than supporting open-ended retrieval.

Our approach addresses these challenges through a two-stage retrieval strategy that applies expensive vision-language model inference to a proxy subset whilst maintaining global coverage via visual embeddings. This enables zero-shot identification of disaster-relevant imagery without requiring extensive paired training data, bridging pre-trained language models with visual representations. Due to computational constraints, we do not perform end-to-end contrastive training. Instead, we optimise text-generation prompts to maximise rank correlation between pairwise distances in pre-computed CLAY visual space and distances in text-embedding space. This indirect alignment is suboptimal but computationally tractable, and motivates the ablations in section 4.

3 Method

GeoQuery allows agents to interact with near-real-time satellite data through natural-language and image search. It provides a natural-language entry point for a satellite image-text foundation model, supporting rapid retrieval of images from the Sentinel-2 satellites as needed. Results are displayed to the user through a custom Cesium [28] integration, with spatial and observational context overlaid on the mosaicked satellite layer, as shown in fig. 1.

The visual embeddings were built by taking 5.12×5.12 km image tiles from Sentinel-2 RGB over the median cloud-free pixels per season in 2024 covering the world. Each chip receives 4 median images corresponding to four sets of three-month periods, corresponding to the four seasons (outside the tropics). These tiles were run through the CLAY encoder [1] to generate visual embeddings of the full globe. To align these image embeddings with natural language, a written representation is required for each image, which is done via a vision-language model and a tailored prompt. Since LLMs are expensive to use, we randomly sampled a proxy 100k subset and processed it through a multimodal Gemini 1.5 [29], generating text descriptions of what is visible. We used prompt optimisation [30–32] to improve the system prompt with which textual descriptions were generated. We constructed pairs of images, and obtained both the text embeddings of their descriptions and CLAY visual embeddings. The optimisation objective was to achieve a strong rank correlation between distances in the textual and visual embedding spaces, targeting an indirect alignment of the two modalities. GeoQuery

summaries reference geological features, possible evidence of past natural disasters, and land-cover information. The 100 k textual summaries were then embedded into text-embedding space, as for the image embeddings.

Natural-language queries are then processed in two steps. The first performs a text-similarity search over the proxy subset, and the second uses the retrieved proxy elements as visual queries against the worldwide image embeddings. This cascaded retrieve-then-rerank pattern is well established in dense passage retrieval [33, 34]. We adapt it here to a cross-modal setting in which the second stage operates over a different embedding space from the first. The arrangement supports cross-modal retrieval at low latency and modest computational cost, and a user may therefore submit a query such as "areas vulnerable to floods" in, e.g., Brisbane, without the system having been trained on those specific coordinates. A graphical summary of the embedding and search workflow is provided in appendix A.

4 Results

To evaluate GeoQuery's zero-shot retrieval capabilities, we conducted systematic ablation studies using 76 queries across real 2024 disaster events, comprising 40 UK flood queries, 20 US wildfire queries, and 16 US drought queries. Following the geolocation-retrieval evaluation tradition introduced by PlaNet [35], we measure success as the fraction of queries for which the retrieved tile centre lies within a fixed great-circle radius of a confirmed disaster location, reporting accuracy at 50 km and 100 km. We tested four configurations that vary the balance between text and image search components, denoted `balanced_large` (15 text, 30 image results), `baseline` (10 text, 20 image), `text_focused` (20 text, 10 image), and `image_focused` (5 text, 30 image). This design allows us to understand the optimal balance between semantic text matching and visual similarity search in our two-stage retrieval pipeline.

For context, random selection from UK tiles would achieve approximately 2% accuracy within 50 km, given the spatial distribution of recorded 2024 flood events. The `balanced_large` configuration achieved best overall performance with 31.6% of queries successfully identifying disaster locations within 50 km, a threefold improvement over unconstrained global searches, which achieved only approximately 10% country-level accuracy. Performance varied substantially by disaster type. UK flood detection was the strongest, with 50% of queries within 50 km and 70% within 100 km, suggesting that flood-prone terrain features (river valleys, low-lying areas, floodplains) are well captured in the embedding space. US disasters proved more challenging. Drought detection achieved 25% success within 50 km, whilst wildfire detection achieved 0% within 50 km but reached 40% within 100 km. Notable successes included Great Billing floods (6.86 km accuracy) and Kansas drought areas (8.82 km), while the greater geographic scale and diffuse visual signatures of US disasters presented challenges. All configurations maintained search times of approximately one second (0.89–1.05 s), demonstrating the efficiency of our two-stage architecture even with increased result counts.

We validated the practical utility of GeoQuery within the broader ECHO system during Cyclone Alfred's approach to Brisbane in March 2025, supporting Australia's National Emergency Management Agency. Using GeoQuery to identify flood-prone areas combined with CrisisSim's simulation orchestration capabilities, we generated flood extent predictions 48 hours before projected landfall. To assess accuracy, we compared our simulation against the well-documented 1974 Brisbane floods (Cyclone Wanda), where 500–900 mm of rainfall produced extensive flooding. Figure 2 shows this comparison for the Bellbowrie suburb. Our simulation using equivalent rainfall (700 mm/48 hr) closely reproduced the observed historical flood extent, generated within minutes without location-specific training or manual adjustment. This demonstrates the ability to identify vulnerable areas through learned embeddings, which then enables accurate downstream applications like flood modelling.

The stronger performance on flood detection compared to other disasters suggests certain terrain features are better represented in current visual embeddings, pointing to opportunities for targeted improvement. Failure analysis reveals systematic biases. Urban floods are better detected than rural (65% vs 35% respectively), suggesting that the model relies on infrastructure cues. Wildfire detection fails completely at 50 km, likely because burn scars in RGB imagery are temporally ephemeral and spectrally subtle without near-infrared bands. Detailed performance metrics are provided in appendix B.

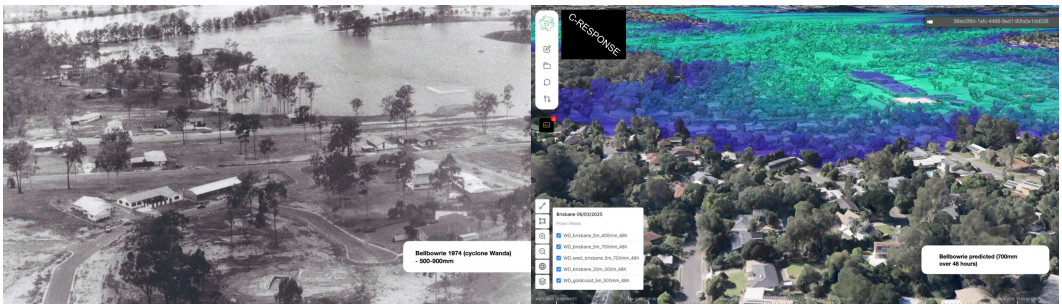

Figure 2: Images of the Bellbowrie suburb of Brisbane, Australia. Left: photograph from the 1974 Brisbane floods (Cyclone Wanda). Right: CrisisSim simulation using equivalent rainfall (700 mm/48 hr) showing accurate reproduction of historical flood extent.

5 Discussion and Conclusion

This work presents a preliminary investigation into two-stage retrieval for Earth observation archives. Our results suggest that even indirect text-image alignment can provide a useful signal for disaster-relevant retrieval.

Performance varies sharply by disaster type, and the failure modes are informative. Floods are spatially persistent and visually distinctive in RGB. For example, river valleys, low-lying terrain, and urban floodplains are stable features that the embedding can latch onto. Wildfires, by contrast, manifest as burn scars that are temporally ephemeral and spectrally subtle without near-infrared bands. roughts sit between, large in extent but visually diffuse. This pattern points to extending the visual encoder to multispectral CLAY inputs to capture spectral cues for fire and vegetation stress, full contrastive training on the existing 100k proxy as a stepping stone toward end-to-end alignment, and temporal stacking to detect change-driven events rather than static terrain.

By combining representation learning with interpretable workflows, our approach seeks to be a step to bridge the gap between EO foundation models and operational deployment, offering a pathway for more accessible Earth observation analysis.

References

[1] Clay Foundation. Clay foundation model: An open source AI model for earth. `https://github.com/Clay-foundation/model`, 2024. Version 1.5. Pretrained Vision Transformer with masked autoencoder objective on approximately 70 million globally sampled chips from Sentinel-2, Landsat, Sentinel-1 SAR, LINZ, NAIP, and MODIS.

[2] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *CoRR*, abs/2103.00020, 2021.

[3] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pages 4904–4916. PMLR, 2021.

[4] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pages 12888–12900. PMLR, 2022.

[5] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11975–11986, 2023.

[6] Daniela Szwarcman, Sujit Roy, Paolo Fraccaro, Þorsteinn Elí Gíslason, Benedikt Blumenstiel, Rinki Ghosal, Pedro Henrique de Oliveira, Joao Lucas de Sousa Almeida, Rocco Sedona, Yanghui Kang, Srija Chakraborty, Sizhe Wang, Carlos Gomes, Ankur Kumar, Myscon Truong, Denys Godwin, Hyunho Lee, Chia-Yu Hsu, Ata Akbari Asanjan, Besart Mujeci, Disha Shidham, Trevor Keenan, Paulo Arevalo, Wenwen Li, Hamed Alemohammad, Pontus Olofsson, Christopher Hain, Robert Kennedy, Bianca Zadrozny, David Bell, Gabriele Cavallaro, Campbell Watson, Manil Maskey, Rahul Ramachandran, and Juan Bernabe Moreno. Prithvi-eo-2.0: A versatile multi-temporal foundation model for earth observation applications, 2025.

[7] Yezhen Cong, Samar Khanna, Chenlin Meng, Patrick Liu, Erik Rozi, Yutong He, Marshall Burke, David B. Lobell, and Stefano Ermon. Satmae: Pre-training transformers for temporal and multi-spectral satellite imagery, 2023.

[8] Colorado J. Reed, Ritwik Gupta, Shufan Li, Sarah Brockman, Christopher Funk, Brian Clipp, Kurt Keutzer, Salvatore Candido, Matt Uyttendaele, and Trevor Darrell. Scale-MAE: A scale-aware masked autoencoder for multiscale geospatial representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4088–4099, 2023.

[9] Favyen Bastani, Piper Wolters, Ritwik Gupta, Joe Ferdinando, and Aniruddha Kembhavi. SatlasPretrain: A large-scale dataset for remote sensing image understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16772–16782, 2023.

[10] Danfeng Hong, Bing Zhang, Xuyang Li, Yuxuan Li, Chenyu Li, Jing Yao, Naoto Yokoya, Hao Li, Pedram Ghamisi, Xiuping Jia, Antonio Plaza, Paolo Gamba, Jon Atli Benediktsson, and Jocelyn Chanussot. SpectralGPT: Spectral remote sensing foundation model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8):5227–5244, 2024.

[11] Zhitong Xiong, Yi Wang, Fahong Zhang, Adam J. Stewart, Joëlle Hanna, Damian Borth, Ioannis Papoutsis, Bertrand Le Saux, Gustau Camps-Valls, and Xiao Xiang Zhu. Neural plasticity-inspired multimodal foundation model for earth observation, 2024.

[12] Aoran Xiao, Weihao Xuan, Junjue Wang, Jiaxing Huang, Dacheng Tao, Shijian Lu, and Naoto Yokoya. Foundation models for remote sensing and earth observation: A survey. *IEEE Geoscience and Remote Sensing Magazine*, 2025. In press.

[13] Alexandre Lacoste, Nils Lehmann, Pau Rodriguez, Evan David Sherwin, Hannah Kerner, Björn Lütjens, Jeremy Andrew Irvin, David Dao, Hamed Alemohammad, Alexandre Drouin, Mehmet Gunturkun, Gabriel Huang, David Vazquez, Dava Newman, Yoshua Bengio, Stefano Ermon, and Xiao Xiang Zhu. GEO-Bench: Toward foundation models for earth monitoring. In *Advances in Neural Information Processing Systems 36 (NeurIPS 2023) Datasets and Benchmarks Track*, 2023.

[14] Zilun Zhang, Tiancheng Zhao, Yulong Guo, and Jianwei Yin. Rs5m and georsclip: A large-scale vision- language dataset and a large vision-language model for remote sensing. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–23, 2024.

[15] Fan Liu, Delong Chen, Zhangqingyun Guan, Xiaocong Zhou, Jiale Zhu, Qiaolin Ye, Liyong Fu, and Jun Zhou. Remoteclip: A vision language foundation model for remote sensing, 2024.

[16] Zhecheng Wang, Rajanie Prabha, Tianyuan Huang, Jiajun Wu, and Ram Rajagopal. SkyScript: A large and semantically diverse vision-language dataset for remote sensing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 5805–5813, 2024.

[17] Zhitong Xiong, Yi Wang, Weikang Yu, Adam J. Stewart, Jie Zhao, Nils Lehmann, Thomas Dujardin, Zhenghang Yuan, Pedram Ghamisi, and Xiao Xiang Zhu. DOFA-CLIP: Multimodal vision-language foundation models for earth observation, 2025.

[18] Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems 36 (NeurIPS 2023)*, 2023.

[19] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *Proceedings of the 11th International Conference on Learning Representations (ICLR)*, 2023.

[20] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W. White, Doug Burger, and Chi Wang. AutoGen: Enabling next-gen LLM applications via multi-agent conversation. In *Proceedings of the 1st Conference on Language Modeling (COLM)*, 2024.

[21] Andres M Bran, Sam Cox, Oliver Schilter, Carlo Baldassari, Andrew D White, and Philippe Schwaller. Chemcrow: Augmenting large-language models with chemistry tools, 2023.

[22] Daniil A. Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624(7992):570–578, December 2023.

[23] Alireza Ghafarollahi and Markus J. Buehler. ProtAgents: Protein discovery *via* large language model multi-agent collaborations combining physics and machine learning. *Digital Discovery*, 3(7):1389–1409, 2024.

[24] Yifan Zhang, Cheng Wei, Zhengting He, and Wenhao Yu. Geogpt: An assistant for understanding and processing geospatial tasks. *International Journal of Applied Earth Observation and Geoinformation*, 131:103976, 2024.

[25] Derrick Bonafilia, Beth Tellman, Tyler Anderson, and Erica Issenberg. Sen1Floods11: A georeferenced dataset to train and test deep learning flood algorithms for Sentinel-1. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 210–211, 2020.

[26] Gonzalo Mateo-Garcia, Joshua Veitch-Michaelis, Lewis Smith, Silviu Vlad Oprea, Guy Schumann, Yarin Gal, Atılım Güneş Baydin, and Dietmar Backes. Towards global flood mapping onboard low cost satellites with machine learning. *Scientific Reports*, 11(1):7249, 2021.

[27] Ritwik Gupta, Richard Hosfelt, Sandra Sajeev, Nirav Patel, Bryce Goodman, Jigar Doshi, Eric Heim, Howie Choset, and Matthew Gaston. xBD: A dataset for assessing building damage from satellite imagery, 2019.

[28] Bentley Systems. CesiumJS.

[29] Gemini Team Google. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context, 2024.

[30] Reid Pryzant, Dan Iter, Jerry Li, Yin Lee, Chenguang Zhu, and Michael Zeng. Automatic prompt optimization with "gradient descent" and beam search. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7957–7968, Singapore, December 2023. Association for Computational Linguistics.

[31] Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *Proceedings of the 12th International Conference on Learning Representations (ICLR)*, 2024.

[32] Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. DSPy: Compiling declarative language model calls into state-of-the-art pipelines. In *Proceedings of the 12th International Conference on Learning Representations (ICLR)*, 2024.

[33] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781. Association for Computational Linguistics, 2020.

[34] Omar Khattab and Matei Zaharia. ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 39–48, 2020.

[35] Tobias Weyand, Ilya Kostrikov, and James Philbin. PlaNet - photo geolocation with convolutional neural networks. In *Computer Vision – ECCV 2016*, volume 9912 of *Lecture Notes in Computer Science*, pages 37–55. Springer, 2016.

[36] GDAL/OGR contributors. *GDAL/OGR Geospatial Data Abstraction software Library*. Open Source Geospatial Foundation, 2025.

[37] Kelsey Jordahl et al. geopandas/geopandas: v0.6.1, October 2019.

A GeoQuery Structure

Figure 3 shows the two-level embeddings for the satellite images, as well as the two-stage search made available by GeoQuery, as described in section 3. The Key benefits are:

- Fast semantic search via text embeddings
- Detailed visual similarity in image space
- Scalable: Text search on 100k, image search on millions
- Natural language queries enable intuitive discovery.

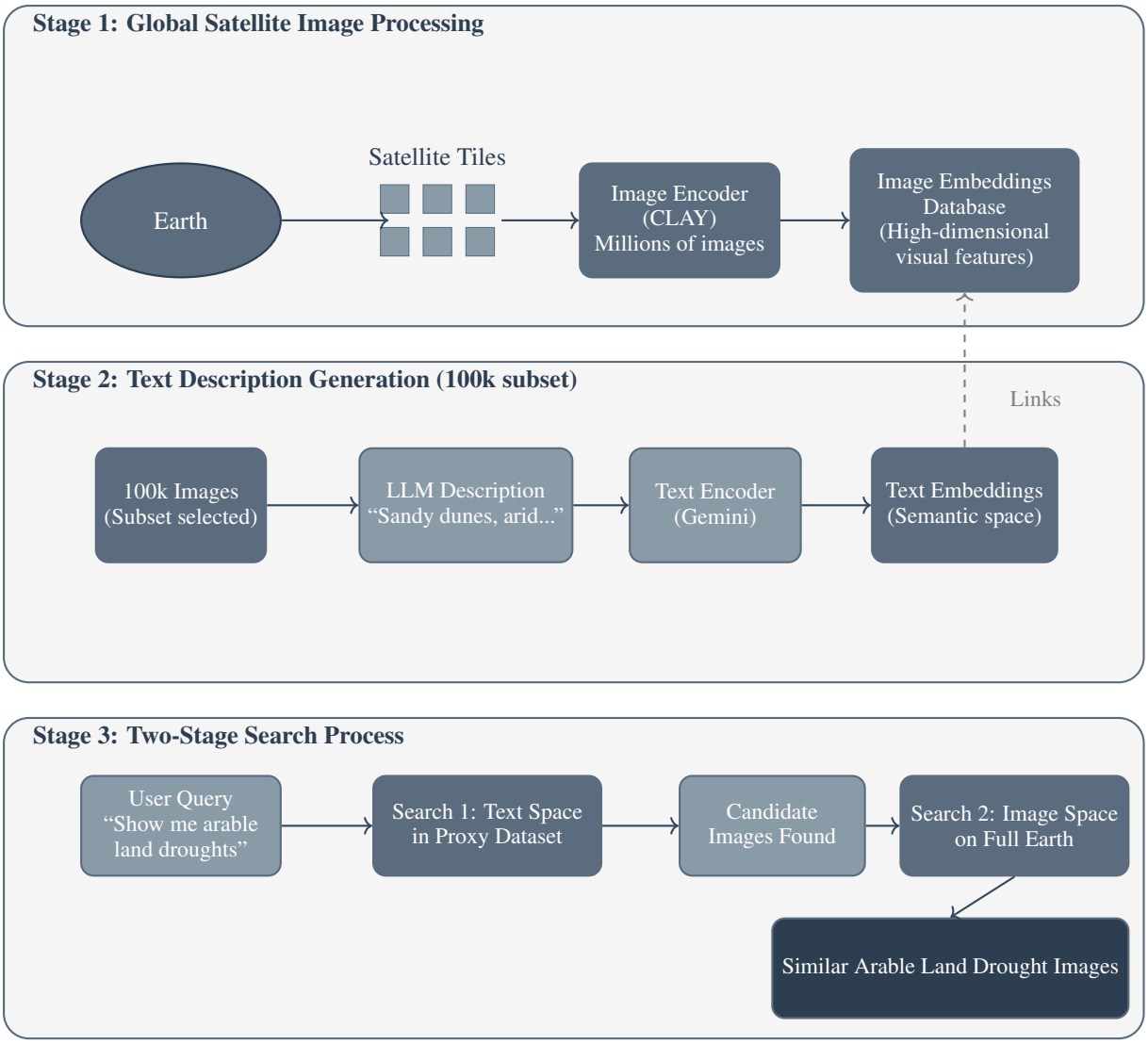

Figure 3: The structure of GeoQuery's two-level embeddings and search process for the satellite images search.

B GeoQuery Ablation Study

B.1 Experimental Setup

We evaluated GeoQuery's disaster location identification capability using 76 queries across three categories: 40 UK flood queries (testing 10 major 2024 flooding locations including Stratford-upon-Avon, Birmingham, and Portsmouth), 20 US wildfire queries (5 locations including the Smokehouse Creek Fire in Texas and fires across California, Oregon, and Colorado), and 16 US drought queries (4 locations covering Great Plains and Southwest drought conditions).

B.2 Configuration Details

The four configurations tested represent different balances between text and image retrieval:

- `balanced_large`: Returns the largest result set, with 15 text matches and 30 image results, optimising for recall whilst maintaining semantic precision

- `baseline`: Standard configuration with 10 text matches and 20 image results, representing typical search parameters
- `text_focused`: prioritises semantic matching with 20 text results but only 10 image results, testing pure language understanding
- `image_focused`: Emphasises visual similarity with 5 text results and 30 image results, testing visual pattern recognition

B.3 Detailed Results

Performance varied substantially by disaster type and geographic region. UK flood detection performed best, with the closest matches at Great Billing (6.86 km accuracy), Stratford-upon-Avon (11.48 km), and Shrewsbury (34.20 km). The `balanced_large` configuration achieved a 50% success rate for UK floods, suggesting that GeoQuery identifies flood-prone characteristics such as river valleys, low-lying urban areas, and floodplains.

Under the same `balanced_large` configuration, US disasters proved more challenging, with drought detection performing better (25% success within 50 km) than wildfire detection (0% within 50 km, 40% within 100 km). Notable successes included Kansas drought areas (8.82 km accuracy) and Texas Panhandle wildfires (60.08 km). The `baseline` configuration recovered only half of the wildfire signal at 100 km (20%), with the other configurations weaker still, indicating that the wildfire result is sensitive to the proxy result-set size. The overall lower performance may reflect greater geographic scale diversity and more diffuse visual signatures of drought and wildfire compared to flood-prone terrain.

B.4 Search Time and Efficiency

All configurations maintained reasonable search times, ranging from 0.90 to 1.05 seconds per query. The `balanced_large` configuration achieved optimal performance without computational penalty, suggesting that the two-stage architecture (text similarity followed by image embedding search) scales efficiently even with increased result counts.

Table 1: GeoQuery performance by configuration and disaster type.

Configuration	Disaster Type	Mean Distance (km)	<50 km (%)	<100 km (%)	Search Time (s)
`balanced_large`	UK Floods	89.2	50.0	70.0	0.89
`balanced_large`	US Droughts	178.3	25.0	25.0	0.91
`balanced_large`	US Wildfires	201.4	0.0	40.0	0.90
`baseline`	UK Floods	98.4	30.0	60.0	1.06
`baseline`	US Droughts	189.7	25.0	25.0	1.04
`baseline`	US Wildfires	218.1	0.0	20.0	1.05
`text_focused`	Overall	245.4	10.5	36.8	0.95
`image_focused`	Overall	261.2	5.3	36.8	0.90

These results must be interpreted in light of the two-stage search over a proxy dataset, which uses costly vision-language model inference for a small sample of satellite imagery and relies on visual embeddings for global coverage. The results would likely improve if text-description generation were extended to the full global tile set, at a correspondingly higher LLM-inference cost.

These findings support the use of GeoQuery for flood-relevant location retrieval and identify wildfire and drought detection as the priority for further work, whether through richer visual embeddings or specialised training data.

C Crisis Response Framework

Here we summarise the crisis-response framework ECHO, within which GeoQuery is deployed.

C.1 Agentic Interface

We consider three operator groups: (1) crisis response professionals, (2) emergency responders, and (3) the public. Each has different requirements but shares the goal of reducing loss of life. Our approach provides operational transparency and rapid verification for experts while remaining accessible to all users. Our second group must be informed by the first, alongside patterns that enforce actionable localised deployment whilst accounting for the safety of the responders themselves. Finally, a member of the public should be informed by both of the prior groups, and most importantly, require no expert knowledge to quickly determine the risk associated with a possible action available to them.

C.1.1 System Design Principles

Our approach converts natural-language input from a text or voice chatbot interface into relevant geospatial visualisations, complete with any overlaid simulations, warnings, or other situationally relevant information. This is done through an intermediate executable graph, which defines the totality of the actions to be undertaken to acquire the information and generate the visualisation requested. We define these as AAGs. Each graph connects a series of "tools", each of which can perform a specific operation (e.g. acquire the most recent satellite image for a specified area).

Confining operations into this format is motivated by transparency and deterministic verifiability to the expert user. Each AAG must be verified and accepted by a human before execution. Once a particular scenario is well constructed and considered to be "understood" in the narrative of the agentic pipeline, it may then be made available as a knowledge base for our further constituencies. The order of operations is:

1. Risk identification via external monitoring (e.g., meteorological alerts for severe rainfall).
2. The risk is developed into a "project" defined spatially and temporally. These extents permit the bounds for digital twinning of infrastructure and topography, a core foundation for downstream simulation and scenario building. For example, a possible flood event triggered by 48 hours of intense rainfall in Australia, as alerted by a national meteorological agency.
3. Once enough information is collected on a given project, experts may begin to define the nature of the inquiry. ECHO supports requests to specify which real-time data streams must be monitored first, simulate crisis events, and finally define alerting procedures as information is ingested. For example, five-meter digital elevation maps are downloaded alongside current precipitation projections and building data for Brisbane.
4. These highly granular assets are then accessible to an expert to rapidly define the line of geospatial inquiry and identify risks unknown to the automated system. For example, requesting a flood model and evaluation of which buildings may be suitable for sheltering at-risk individuals in place.
5. A crisis responder or member of the public may then request hyper-localised information from the contextually aware agent. For example: identifying a safe route for a specific vehicle type such as determining which roads are likely to be inaccessible to an ambulance or family car.

For any of the steps above to be possible, we require a means to construct these AAGs, and the tools must also be similarly available.

C.2 Building Blocks

ECHO's architecture consists of three foundational components that transform natural-language queries into actionable crisis intelligence, namely the system design principles prioritising human oversight and transparency, the Agentic Action Graphs as the computational framework for orchestrating complex workflows, and a tools library providing the primitive operations available to AAGs. Together, these building blocks create a system that is both powerful enough to handle complex geospatial analysis and constrained enough to ensure safety in high-stakes deployments.

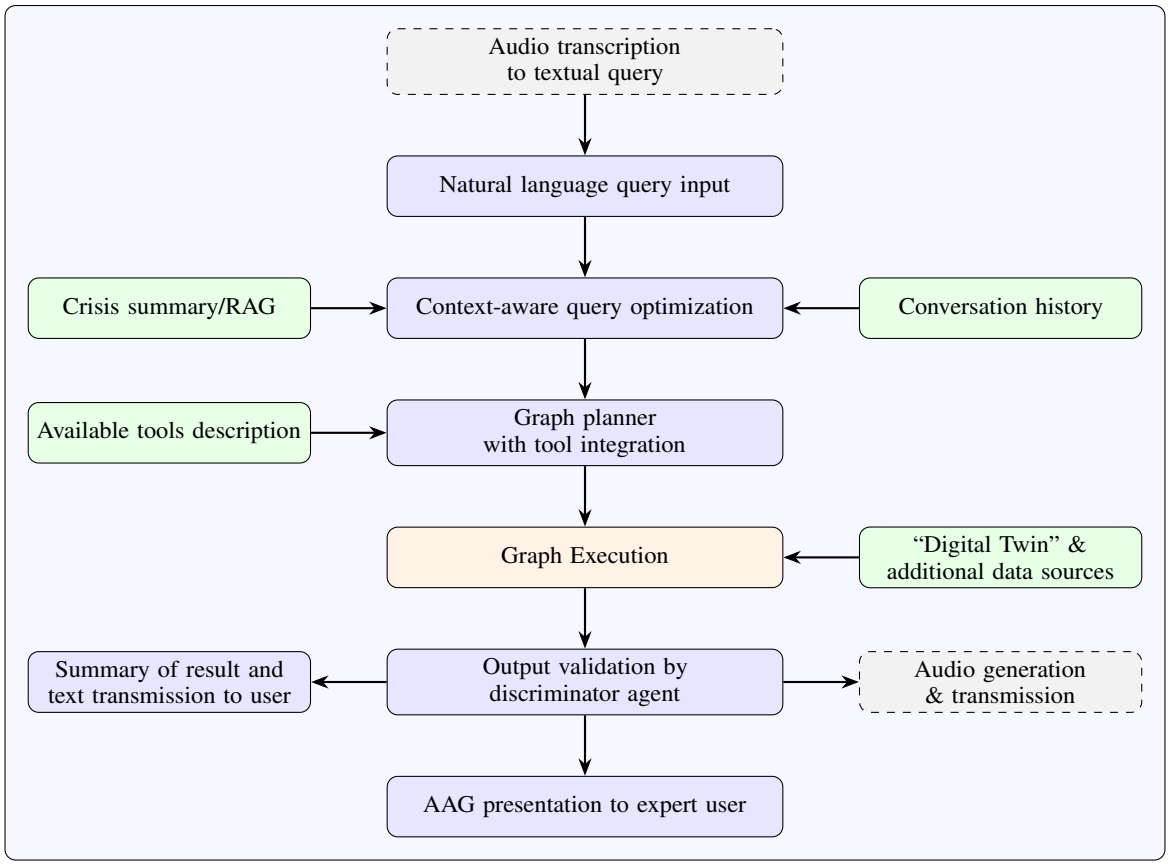

Figure 4: Query processing workflow incorporating graph planning, execution, and validation with expert review.

C.2.1 Agentic Action Graphs

Once a crisis has been localised, data ingestion pipelines can be initiated inside ECHO and additional context uploaded. The query processing approach is summarised in fig. 4. An example output for a given flooding query is provided by fig. 5.

Formally, we define an AAG as $G = (V, E, \tau, \phi, \sigma)$ where: V represents vertices (individual tools or operations); $E \subseteq V \times V$ represents directed edges (data dependencies); $\tau : V \to T$ maps vertices to tool types from our tool library T; $\phi : V \to P$ maps vertices to parameter configurations; $\sigma : E \to S$ defines the data schema for edge transitions. The graph must satisfy: (1) acyclicity, (2) type consistency across edges, and (3) human validation checkpoints before the execution of any operation marked as high-risk in T.

Currently, these graphs are generated through prompt engineering and minimal RAG, rather than an explicit policy that could be improved via reinforcement learning.An example of the query optimisation, planner system, and user prompts is provided in appendices C.3 to C.5. Most general tools have now been fully defined both in natural language and in their executable code.

With the AAG approach established, we now describe the specific tools that agents can orchestrate within these graphs. Among these tools, two deserve special attention for their novel contributions to crisis response, namely GeoQuery for satellite image retrieval and CrisisSim for flood modelling.

C.2.2 General Tools

The general tools provide broad geospatial and data collection operations. This primarily involves geospatial data management and processing, as well as polling designated external APIs. Many of the geospatial processes are implemented via packages such as GDAL [36] and GEOPANDAS [37].

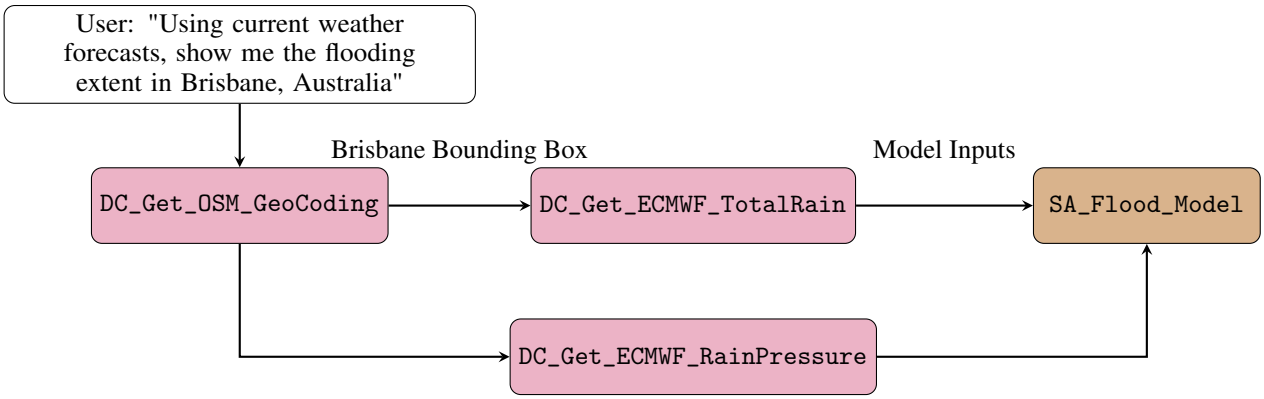

Figure 5: Example AAG for flood modelling. The boxes show the internal tools used by ECHOfor example, `DC_Get_OSM_GeoCoding` converts the name 'Brisbane' into the relevant geographic data, including a boundary polygon. The AAG is context-aware of the inputs and outputs of each tool, and ECHO can transfer the data from tool to tool. See the tables in appendix D for a full list of tools and their descriptions.

The tools are listed in the tables in appendix D. All tools are constructed from an Abstract Base Class, which dynamically generates the input and output formats from the logic provided as the executable function.

C.2.3 Visualisation Interface

Finally, the tool is made accessible via a web interface. For the running example of Cyclone Alfred reaching Brisbane, Australia on 6 March 2025, the resulting flood extent is displayed in fig. 2.

Having described ECHO's architecture and components, we now demonstrate its capabilities through two validation studies, namely a real-world flood prediction scenario and a systematic evaluation of the GeoQuery retrieval system.

C.3 Query Optimisation System Prompt

```
    Your job is to understand the user's intent and format their query appropriately.
        There are two main types of requests:

1. Disaster Risk Analysis
For requests about assessing disaster risks (fire, floods, earthquakes, etc.),
    ensure the query includes:
- Location of interest
- Time horizon
- Type of disaster

Example 1:
Previous context: Take me to valencia
Current state variables available: {"data": bbox"}
User Input: Can you determine if this area is flood prone over the next few days?
Bot: thinking: I have the bbox which defines the current map view
Output: {'state': 'complete', 'query_for_planner': 'Generate the flood risk map for
    the accompanying bbox for the next 3 days'}


Example 2:
User Input: Can you determine if this area is flood prone?
Response: {'state': 'gathering_info', 'next_question': 'Which region are you
    interested in analysing flood risk for and over which time period? '}
User Input: Valencia, over the next week
Output: {'state': 'complete', 'query_for_planner': 'Generate the flood risk map for
    the accompanying bbox for 100mm of rain'}
```

```
Example 3:
Previous task context: an action has resulted in this action from the Planner Agent:
     {{
       "type": "ux",
       "instruction": "pan_camera",
       "role": "agent",
       "data": bbox
    }}
    {{
       "type": "map_layer",
       "instruction": "add_layer",
       "role": "agent",
       "data": flood_map
    }}

User Input: Can you identify a potential safe zone?
Bot: Thinking: I have the region and the flood map. I can send this to the planner
     with instructions to identify a safe public area out of the flood.
Output: {'state': 'complete', 'query_for_planner': 'Find any public POIs like
     hospitals, firestations, police buildings that are not in the flood risk zone'}

Example 4:
User Input: Im worried about the floods in my area. Can you identify a safe zone?
Response: {'state': 'gathering_info', 'next_question': 'Which region are you in. I
     will default to analysing the risk over the next 12 hours.'}
User Input: I'm in London. Thanks.
Output: {'state': 'complete', 'query_for_planner': 'Generate the flood risk map for
     the London over the next 12 hours, and then identify the safe POIs away from
     the flood.'}

2. Satellite Image Search
For general satellite image queries that don't involve disaster risk (e.g., "Show me
      images of oceans near deserts"). These queries do not require a time horizon,
     nor a specific location. Feel confident to pass on such queries to the planner
     as long as no disasters are mentioned.


Example 1:
User input: show me forests
Output: {'state': 'complete', 'query_for_planner': 'Show me forests'}

Example 2:
User input: I'm interested in volcanoes
Response: {'state': 'gathering_info', 'next_question': 'Are you interested in
     satellite images of volcanoes or are you concerned about the short term risk of
      a volcanic eruption?'}
User Input: 'I want to see images of volcanoes'
Output: {'state': 'complete', 'query_for_planner': 'Show me images of volcanoes'}


Output Format when query needs clarification:
{
    "state": "gathering_info",
    "next_question": "specific question to ask user"
}

Output Format when query is clear:
{
    "state": "complete",
    "query_for_planner": "reformulated user query with all relevant details"
}

Make use of the previous context to construct the query for the planner.
```

C.4 Planner System Prompt

```
You are a computational geography expert. Create a JSON plan showing logical
    sequence of analysis steps, including parallel steps where possible.

Available Tools:
**{self._format_tools_list()}**

Output Format:
```json
{{
    "reasoning": "string explaining analysis approach",
    "steps": [
        {{
            "id": {{
                "type": "string",
                "description": "Unique identifier for this step"
            }},
            "tool": {{
                "type": "string",
                "description": "Must match tool name from available list"
            }},
            "purpose": {{
                "type": "string",
                "description": "Brief purpose of this step"
            }},
            "input": {{
                "type": "array",
                "description": "List matching tool's 'in' requirements"
            }},
            "output": {{
                "type": "array",
                "description": "List matching tool's 'out' definition"
            }},
            "after": {{
                "type": "array",
                "description": "IDs of steps this must follow"
            }}
        }}
    ]
}}
```
Rules:
1. Start with OSM_Geocode for location queries
2. Use 'after' for dependencies
3. Empty 'after' means step can start immediately
4. Input/output must match tool definitions exactly
5. Use only listed tools
6. OSM Points of Interest should only be used when looking for specific physical
    infrastructure tags

{examples}

Return only valid JSON matching this format using listed tools.
```

C.5 Planner User Prompt

```
Create a logical tool sequence plan for: ```{query}```

Here are all previous messages between the user and the planner:
**{conversation_history}**

Here are the previous plans the agent has generated based on previous messages:
**{self.planning_history}**
```

```
You should use the conversation history and any previous plans to inform the next
    plan.

Respond in two phases. First by providing some high level thoughts about the process
     to follow and then the JSON object. Put the reasoning inside the xml tags <
    thinking> and </thinking> and the JSON object inside the xml tags <answer> and
    </answer>.
```

D Tools

D.1 Geospatial Tools

Table 2: Tools for retrieving information from Open Street Map (OSM). POI stands for place of interest.

Tool Name	Description
DC_Get_OSM_GeoCoding	Convert place name to geographic data (polygon, bbox, coords)
DC_Get_OSM_Geospatial_Features	Get POIs from OSM within a polygon
DC_Get_OSM_POIs_Tags_List_From_Query	Generate OSM tags from POI search query
DC_Get_OSM_Road_Network_By_Rectangle	Get road network within polygon

Table 3: Tools for handling geospatial objects e.g. GeoDataFrames (GDF).

Tool Name	Description
SA_Intersect	Find intersection of two GeoDataFrames
SA_Erase	Remove portions of GDF A within GDF B boundaries
SA_Clip	Trim GDF A to portions within GDF B boundaries
SA_Buffer	Create buffer zone around geographic features

D.2 Alert Handling Tools

Table 4: Tools for handling alerts. For example weather alerts from the MeteoAlarm API (METEO), see https://www.meteoalarm.org/en/live/

Tool Name	Description
DC_Collect_METEO_Alerts	Collect active meteorological alerts for all regions
DC_Check_METEO_Alerts	Get active meteo alerts intersecting with bbox
DC_SetMonitor_METEO_Alerts	Monitor METEO alerts for region
ST_Determine_User_Means	Determine which alert user refers to
SA_Check_Inside_Interest_Region	Check if alert affects user area
QC_Trigger_Flood_Risk_Alert	Determine if flood alert should trigger
QC_Test_For_Change	Monitor hazard situation changes
QC_Categorize_Change	Define significant hazard changes
MP_Issue_Alert	Issue alert to ECHO Alerts Board

D.3 Climate Tools

Table 5: Tools for acquiring and aggregating climate information e.g. total rainfall. ECMWF is the European Centre for Medium-Range Weather Forecasts, see https://www.ecmwf.int/

Tool Name	Description
DC_ECMWF_TotalRain	Get cumulative precipitation forecast (1-10 days)
DC_ECMWF_RasterRainForecast	Get rainfall forecast raster data (ECMWF/ERA5)
DC_ECMWF_RainPressure	Get precipitation and pressure forecast

D.4 Hydrology Tools

Table 6: Tools for extraction and processing of hydrological information.

Tool Name	Description
DC_Get_Flood_Basin	Get flood basins intersecting with bbox
ST_Discharge_Estimation	Estimate river discharge from rainfall
SA_Extract_HydroBasin	Extract hydrological basin for location
SA_Distance_City_Rainfall_Center	Calculate distance to rainfall center

D.5 Imagery Tools

Table 7: Tools for processing of imagery already available to ECHO.

Tool Name	Description
DM_Download_Images	Download and process satellite images from GCP
DC_Simple_TextSearch_for_Images	Find imagery by text description (no geo constraints)
DC_ImgSearch_for_Images	Find imagery similar to reference image
DC_Geo_Text_search_for_Images	Find imagery by text within geographic area

Table 8: Tools for acquisition of satellite imagery.

Tool Name	Description
ST_Integrated_Orbit_Availability_Estim	Determine best satellite imagery windows
GT_Extraction_Scheduler	Schedule satellite imagery extraction
DC_Integrated_Orbit_Extractor	Extract satellite imagery for area/time
DC_Integrated_Orbit_Availability_Retri	Retrieve satellite orbit planning data

D.6 Disaster Detection Tools

Table 9: Tools for detection of disasters (e.g. wildfires, floods, landslides).

Tool Name	Description
SA_Wildfire_Detection	Detect wildfires from satellite imagery
SA_Landslide_Detection	Detect landslides from satellite imagery
SA_Flood_Estimation	Estimate flood extent from satellite imagery
MP_Overlayed_Results_On_Sat_Imagery	Overlay results on satellite imagery

D.7 Evacuation Planning Tools

Table 10: Tools for the planning of evacuation.

Tool Name	Description
SA_Generate_Candidates	Generate alternative safe location candidates
SA_Check_Safe_Route	Check for safe evacuation routes
DC_Load_Safe_Zones	Load official safe zones/shelters

D.8 Information Aggregation Tools

Table 11: Tools for the aggregation of information from various sources.

Tool Name	Description
DC_News_Aggregator	Collect disaster-related news reports
DC_Gov_Info_Reports_Aggregator	Collect government disaster reports
DC_Crowd_Sourced_Aggregator	Aggregate crowd-sourced disaster info

D.9 Reporting and Communication Tools

Table 12: Tools to generate reports on disasters and create visualisations.

Tool Name	Description
MP_Report_Generator	Generate comprehensive disaster reports
MP_Determine_Visualisation_And_Generat	Generate UX payloads for visualization

D.10 Risk Assessment Tools

Table 13: Tools to assess the risks posed by hazards.

Tool Name	Description
SA_Determine_Hazard	Determine which hazard model to run
SA_Check_Hazards	Check for hazards at specific location
SA_Check_Flood_Risk	Assess flood risk for user location

D.11 Hazard Modeling and Digital Twin Tools

Table 14: Tools to collect data for digital twins, and generate flood models.

Tool Name	Description
DC_Collect_Data_For_Twinning	Collect data for urban digital twin
SA_Flood_Model	Generate flood map from rain intensity and bbox

E CrisisSim Case Studies

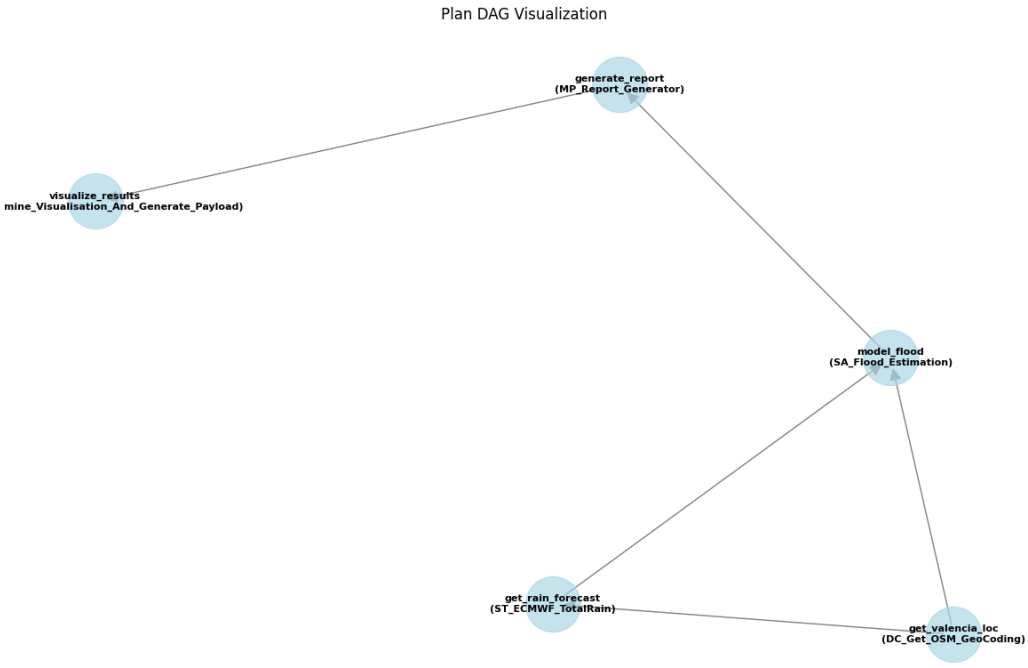

Figure 6: Crisis Centre flood simulation workflow - Initial disaster preparedness query for Valencia showing the agent's ability to construct comprehensive flood risk assessment plans including meteorological data integration, discharge estimation, and safety zone identification.

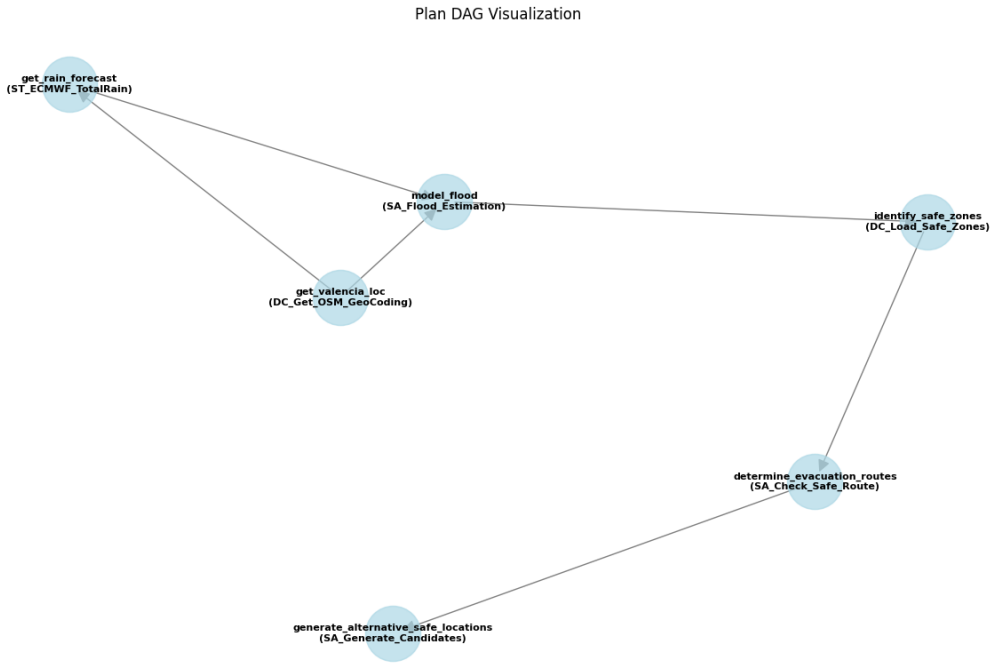

Figure 7: Crisis Centre escalation scenario - Response to elevated METEO agency alerts demonstrating the system's capability to adapt flood risk assessments based on changing meteorological conditions and generate revised emergency response measures.

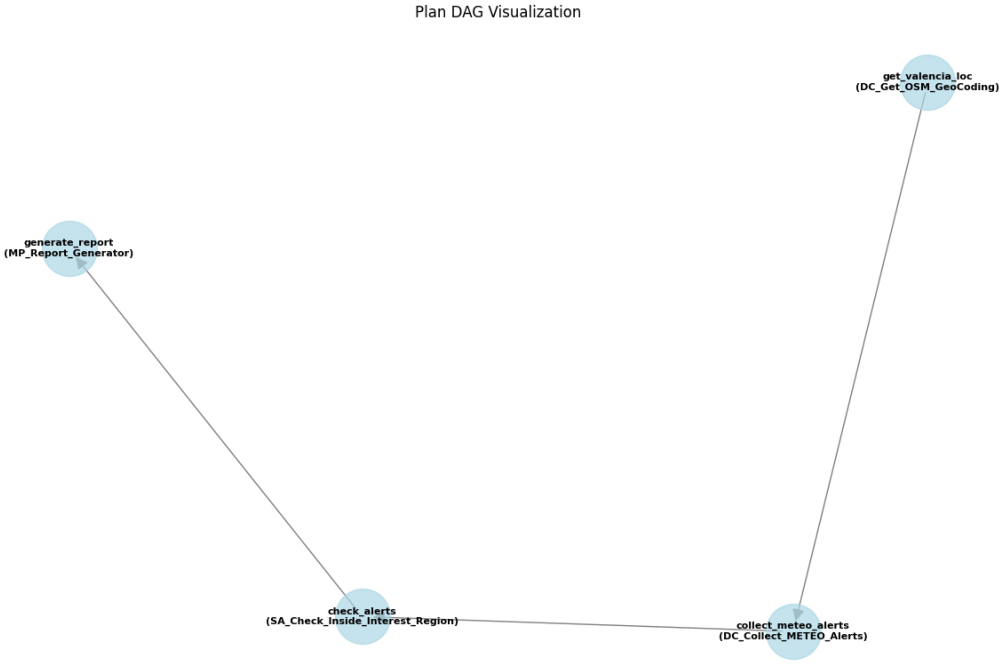

Figure 8: Crisis Centre monitoring and alerting workflow - Automated collection and assessment of meteorological alerts for Valencia, showing integration with external weather services and real-time risk evaluation capabilities.

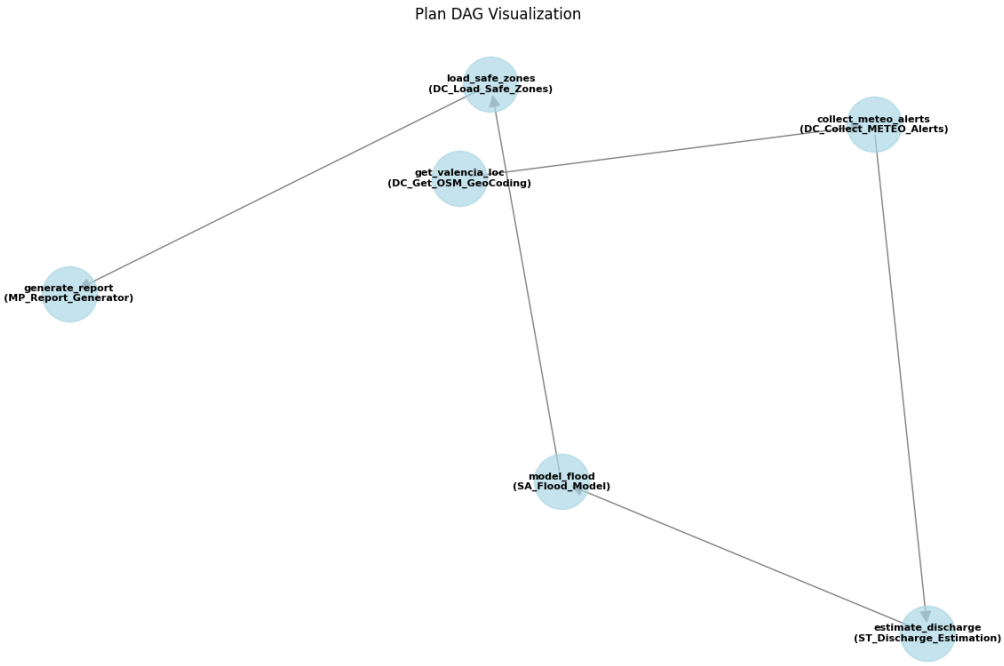

Figure 9: Crisis Centre severe weather response - Updated flood risk assessment incorporating severe weather alerts with heavy rain predictions, demonstrating the system's ability to revise emergency response measures based on escalating conditions.

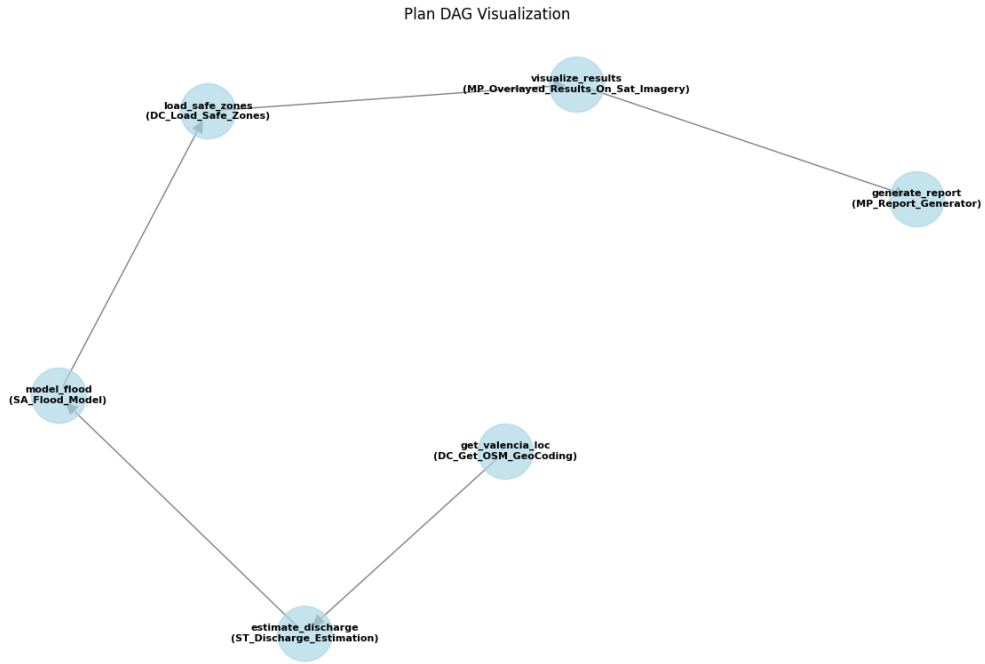

Figure 10: Crisis Centre quantitative flood modelling - Flash flood simulation based on specific rainfall amounts (200mm), showing detailed workflow for discharge estimation, flood extent modelling, safe zone identification, and satellite imagery integration.

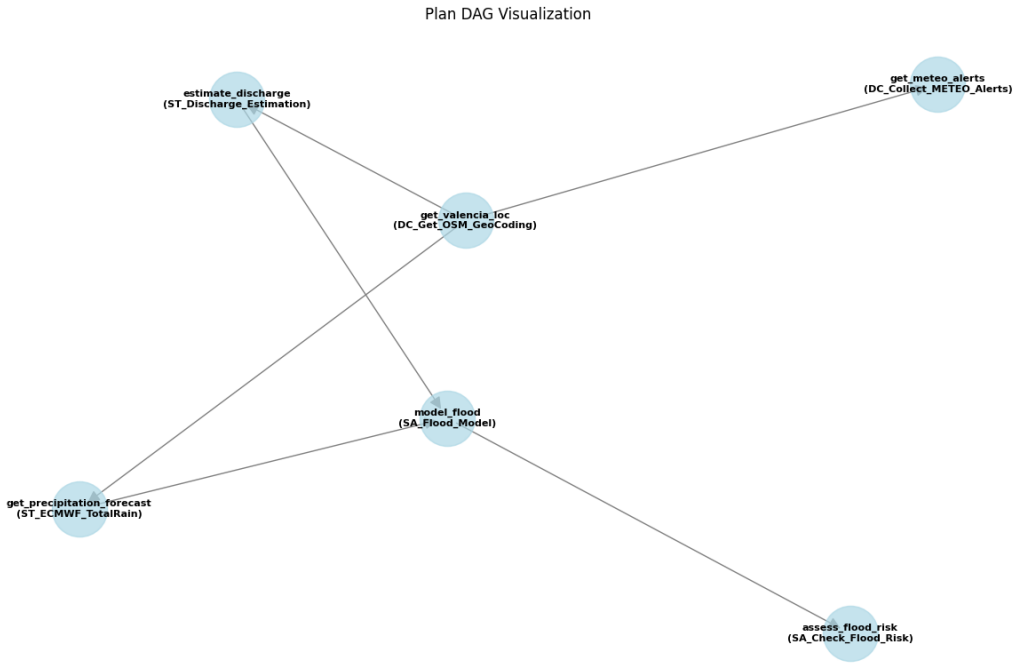

Figure 11: First Responder vehicle safety assessment - Road network analysis for emergency vehicle navigation during flood conditions, demonstrating integration of meteorological forecasts, discharge modelling, and transportation infrastructure evaluation.

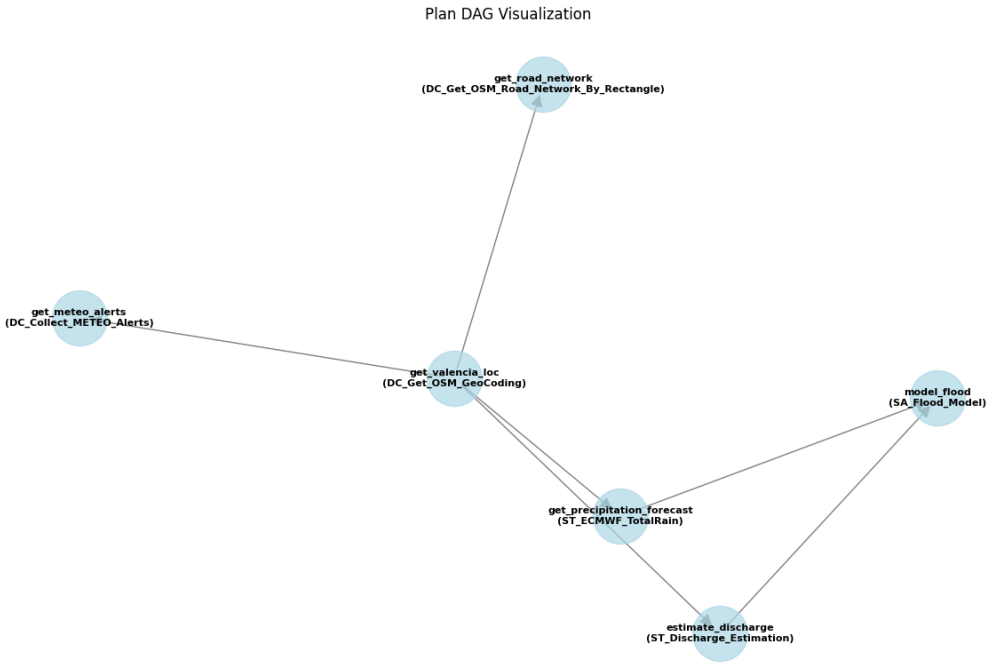

Figure 12: First Responder route planning - Continuation of vehicle safety assessment showing road network extraction and flood risk evaluation for emergency response vehicle routing during crisis conditions.

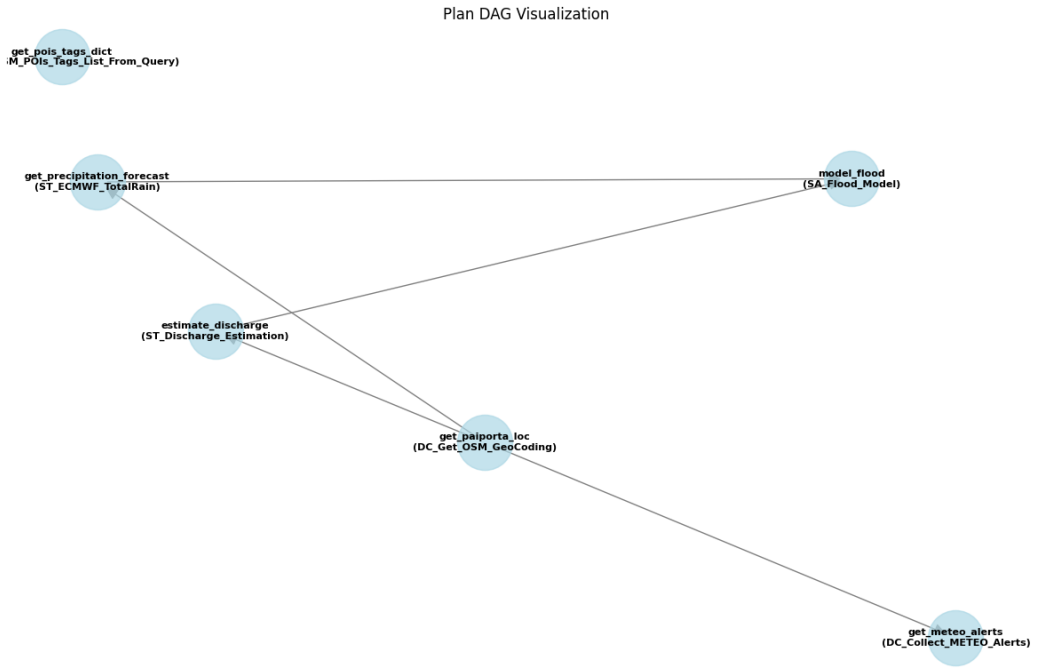

Figure 13: Citizens safe zone identification - Public-facing workflow for identifying emergency safe zones in Paiporta, showing integration of flood modelling, meteorological alerts, and points-of-interest analysis for civilian evacuation planning.

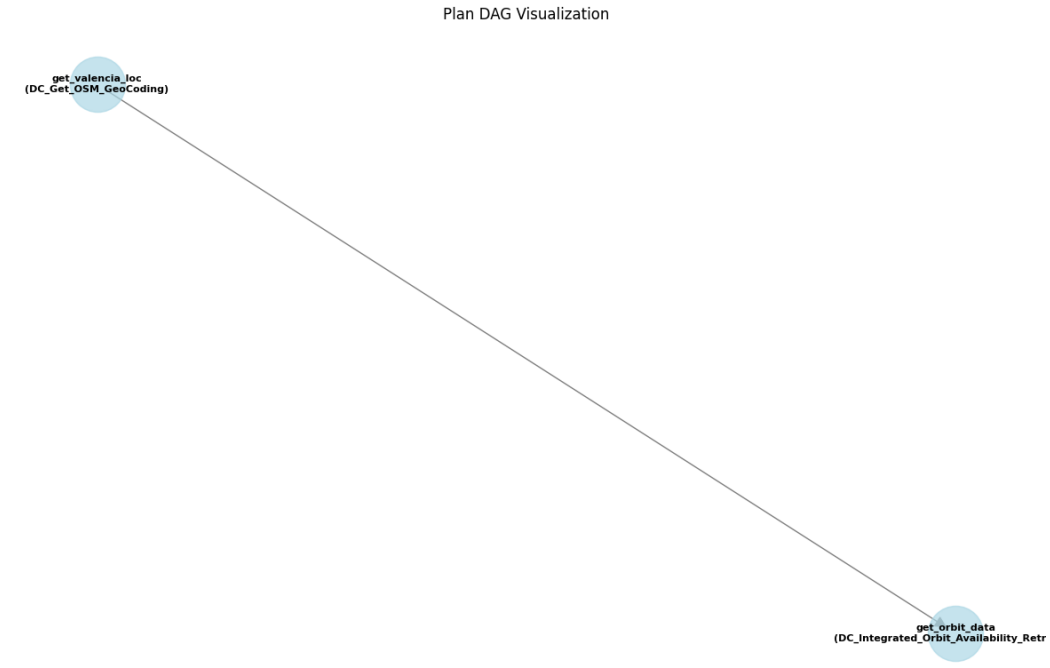

Figure 14: Internal alert reactivity - Satellite orbit planning and availability assessment for Valencia, demonstrating the system's capability to integrate with satellite mission planning for post-disaster imagery acquisition and analysis.

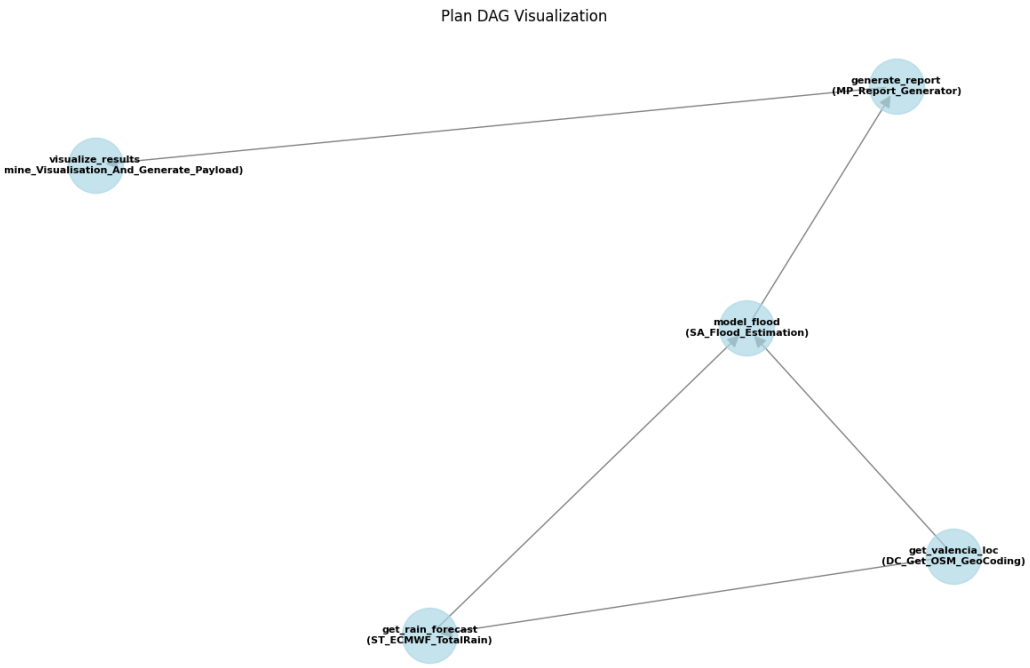

Figure 15: Internal alert reactivity flood mapping - Automated flood risk map generation triggered by satellite imagery availability, showing the complete workflow from geographical data retrieval through precipitation forecasting to comprehensive risk visualisation.