We thank the reviewers for their detailed and thoughtful comments. We are happy to see that reviewers found our paper has ‘strong significance’ and ‘novel insight into map architectures’ (R1), ‘provides a clear signal on how to think about mapping and navigation’ (R2), gives ‘tasks that are interesting and clean’ and an ‘extensive study of mapping capabilities’ (R3), and is ‘an important work, demonstrating several useful and interesting findings’ (R4). We answer questions by reviewers below and will add related feedback with suggested references and clarifications in the paper.

R1, R2, R3: Other agent memory architectures? Our focus was to analyze the impact of information in map representations. We agree that discussing and benchmarking more agent architectures will strengthen our findings. Following reviewer suggestions, we conducted experiments with SMT [17], EgoMap [6] and FRMQN [Oh et al., 2016]. Results are consistent with our earlier findings, with all three performing better than the simple RNN memory in `NoMap` but worse than `ObjRecogMap`, with differences being more pronounced on 3-ON. As suggested by R2, we will add a discussion of the choice of agent architecture and memory representation, including Banino et al.

	Success (%)			Progress (%)			SPL (%)			PPL (%)		
	1-ON	2-ON	3-ON	1-ON	2-ON	3-ON	1-ON	2-ON	3-ON	1-ON	2-ON	3-ON
`NoMap` (RNN)	62	24	10	62	39	24	35	13	4	35	21	14
`ProjNeuralMap`	65	30	12	65	44	27	37	20	5	37	28	14
`ObjRecogMap`	**79**	**51**	**22**	**79**	**62**	**40**	**56**	**38**	**17**	**56**	**45**	**30**
`SMT` [17]	63	28	9	63	44	22	48	26	7	48	36	18
`EgoMap` [6]	66	36	16	66	48	32	49	27	12	49	35	24
`FRMQN` [Oh et al., 2016]	62	29	13	62	42	29	50	24	11	50	33	24

R1: Measure variance with multiple seeds? We did not include this analysis in the paper since early experiments showed low training variance, and reporting results with one training seed is common in embodied AI for navigation [11,17,21-23]. The standard deviation across 5 training runs on 3-ON of the `OracleMap` and `ProjNeuralMap` agents is 0.38% and 0.72% for SPL, and 0.52% and 0.99% for PPL (all within 1%).

R2: Human-readable top-down maps are not necessarily optimal for learning-based agents. Top-down maps are indeed one of many possible spatial representations. They are a good starting point as they are common in real life, and humans are taught to understand and use them reliably. We do not mean to imply they are optimal for learning-based systems. Top-down maps do bring the advantage of a good abstraction, human interpretability, and they impose inductive biases tied to the spatial structure of interiors that have been shown to outperform implicit memory architectures in a variety of navigation tasks [6,11,12]. We will clarify this in the paper.

R3,R4: Hardness of the task as m increases and allotted time limit. Reviewers asked whether it is surprising that agents fail to perform well with increasing m as: 1) the task is obviously exponentially harder, and 2) there is a fixed max time limit. We clarify here and will add discussion of these points to the paper. **R3: Obvious that task is exponentially harder?** With exponential decay, we expect the `OracleMap` Success rate to be 0.94, $0.94^2 = 0.88$ and $0.94^3 = 0.83$ as m is increased from 1 to 3. The actual Success rate is much lower indicating room for improvement. In addition, if the agent had the ability to remember past observations, finding the third goal should be easier if it was encountered while looking for the first two goals. This is supported by the analysis in Section 1.4 of the supplement. **R4: Is max time limit independent of m in m-ON?** Yes, we set the max number of steps for all tasks to 2500 steps. Nearly all (> 99.5%) episodes terminate by calling the FOUND action rather than by reaching this limit. The mean and median episode lengths being 276.2 and 151 steps respectively for 3-ON experiments. We verified that this fixed max step threshold has negligible impact on the results reported in the paper.

R3: How much is episode terminating on incorrect FOUND making the problem harder? Yes, the FOUND action makes the problem much harder. Without it, the task is reduced to a search problem bounded by the max step limit. We evaluated an `OracleMap` agent with ‘OracleFound’ (FOUND called automatically). This achieves Success of 90% in 3-ON, compared to 48% for `OracleMap` that must call FOUND. The FOUND action follows the recommendation of [2] and is a realistic requirement: it indicates the agent correctly identified the object and is ready to move on to the next goal. In initial experiments, we trained agents with the episode not ending on incorrect FOUND. These agents fail to learn, issuing FOUND randomly and wandering in an episode until the max step limit is reached.

R2: Unusual ‘r_closer’ (reward for coming closer to goal). In early experiments without this reward, agents failed to train even after 40 million steps. This form of reward shaping is commonly employed in RL-based embodied navigation [31,37], and is crucial for facilitating training as a weak supervisory signal that mitigates sparse rewards.

Other clarifications. R3: Global position of agent is not part of the map? and R4: Oracle maps appear to be in allocentric coordinates? As input to the agent, all map-equipped agents receive maps centered and rotated at the agent’s location and orientation. R4: Taxi/courier task an important precedent. Thank you for the suggestion, we will add this work to the discussion. R2,R3: We will clarify the exposition for L122, L126, L167-170, L201, and L207.