



Automatic Undo for Cloud Management via AI Planning

Ingo Weber, Hiroshi Wada, Anna
Liu, Alan Fekete, and Len Bass



Australian Government
Department of Broadband, Communications
and the Digital Economy
Australian Research Council

NICTA Funding and Supporting Members and Partners

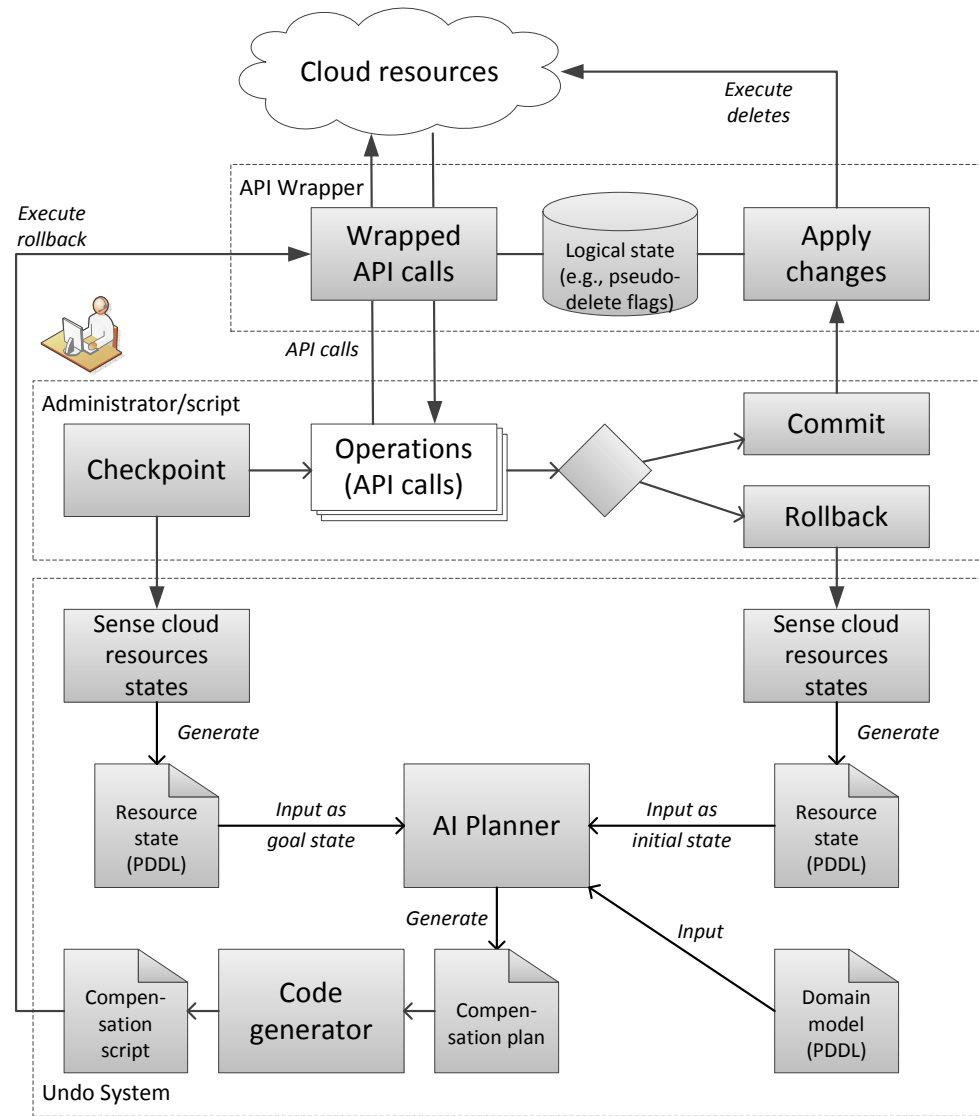


Motivation



- Audience of approach: users of Cloud Mgmt APIs
- After trying out something in AWS, may want to go back to original state
 - Encountered during product development for Yuruware
 - No unit testing possible – annoying to undo tests
 - If something fails, resources are left in arbitrary states
- Reverting not always that straight-forward:
 - Attaching volume is no problem while the instance is running, detaching might be problematic
 - Creating / changing auto-scaling rules has effect on number of running instances
 - Cannot terminate additional instances, as the rule would create new ones!
 - Deleted / terminated / released resources are gone!

Approach



Example



- State in AWS:
 - Instance i1 running
 - Elastic IP e1 associated with i1
- Do:
 - terminate i1
- Undo
 - undelete i1
 - start i1
 - associate e1 to i1

Why AI Planning?



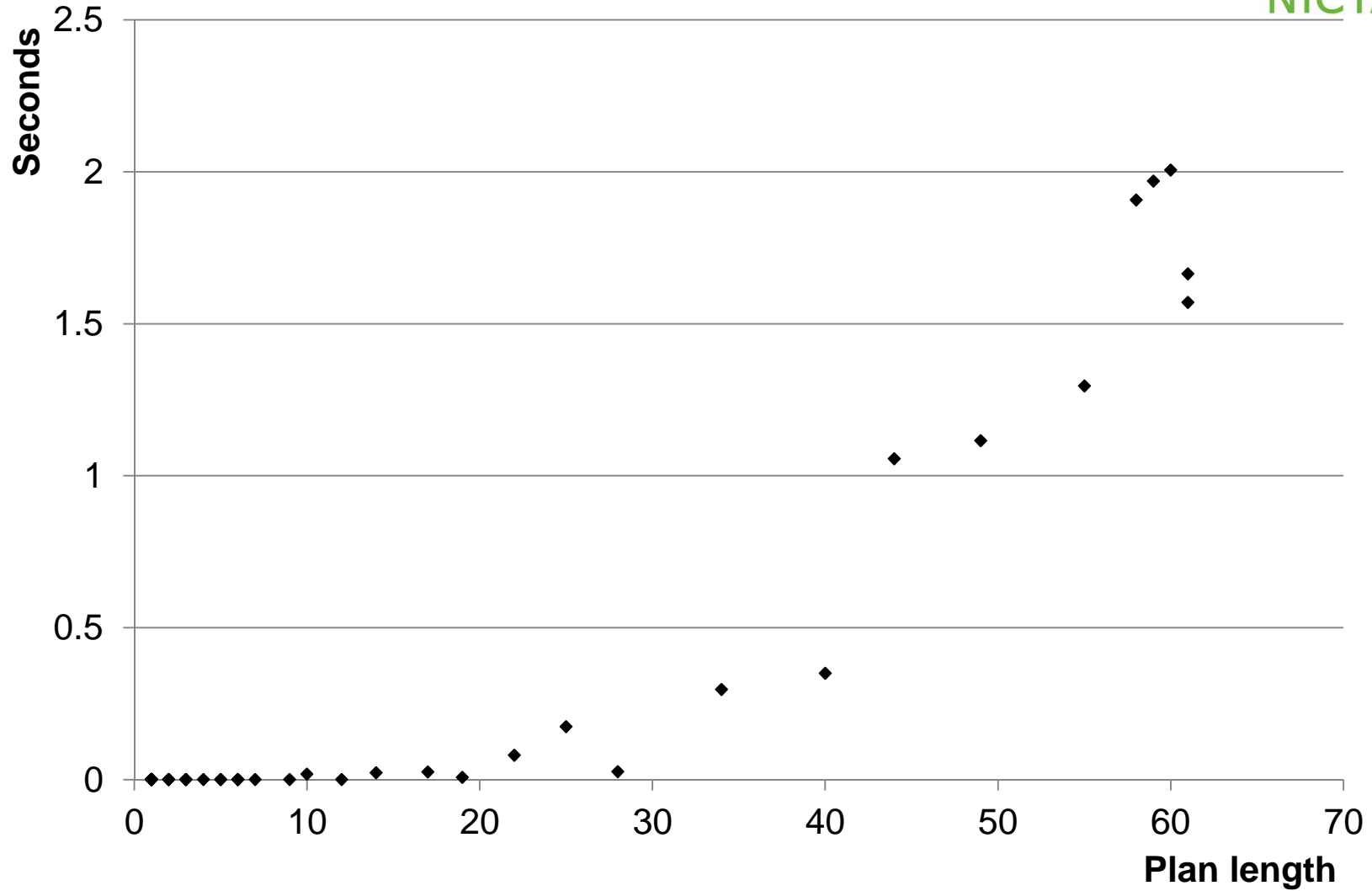
- Traditional techniques to rollback long-running transactions do not apply or are sub-optimal:
 - *Sagas* (execute inverse ops in reverse chronological order), does not work on Amazon Web Services
 - Hand-coding handling for all possible cases is tedious
- AI Planning:
 - Given start state, goal state, set of actions, searches a solution in the state of possible plans
 - Highly optimized heuristics tame the PSPACE-hard problem for practical purposes
 - Our variant finds ,maximal‘ contingency plans
 - if one action fails, but the goal is still reachable, a backup plan is found

Evaluation



- Basis: prototype
 - full implementation (for selected resource types);
planner-only implementation (for more resource types)
- Use cases: ~70 different planning settings tested
- Performance 1: scaling plan length
- Performance 2: scaling number of unrelated resources

Evaluating performance 1: plan length



Conclusions & future work

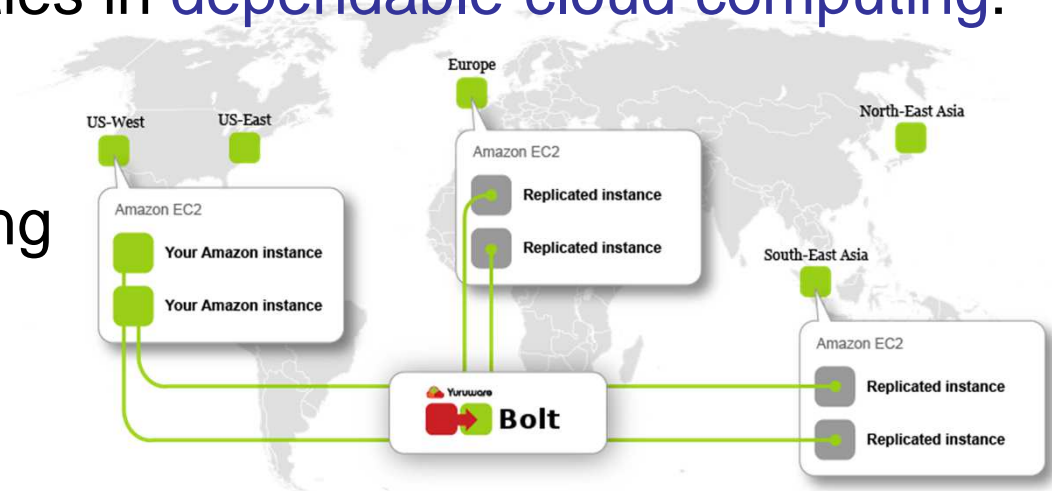


- Summary
 - Approach and prototype for rollback in cloud management, using AI Planning techniques
 - Formalized part of AWS APIs in a planning domain model
 - Used an off-the-shelf planner and developed a prototype around it
 - Scales well in terms of number of rollback operations needed, for practical system sizes
- Future work
 - Finding forward plans / “do”
 - Parallelizing plans
 - Extending checkpoints to capture internal resource state

Thank You!

Research study opportunities in **dependable cloud computing**:

- Software Architecture
- Data Management
- Performance Engineering
- Autonomic Computing



To find out more, send your CV and undergraduate details to
students@nicta.com.au