Prov-DIFF: Play traces analysis through provenance differences

Authors: **Troy C. Kohwalter,** Leonardo G. P. Murta, Esteban W. G. Clua

{troy, leomurta, esteban}@ic.uff.br

Instituto de Computação – Universidade Federal Fluminense (UFF)

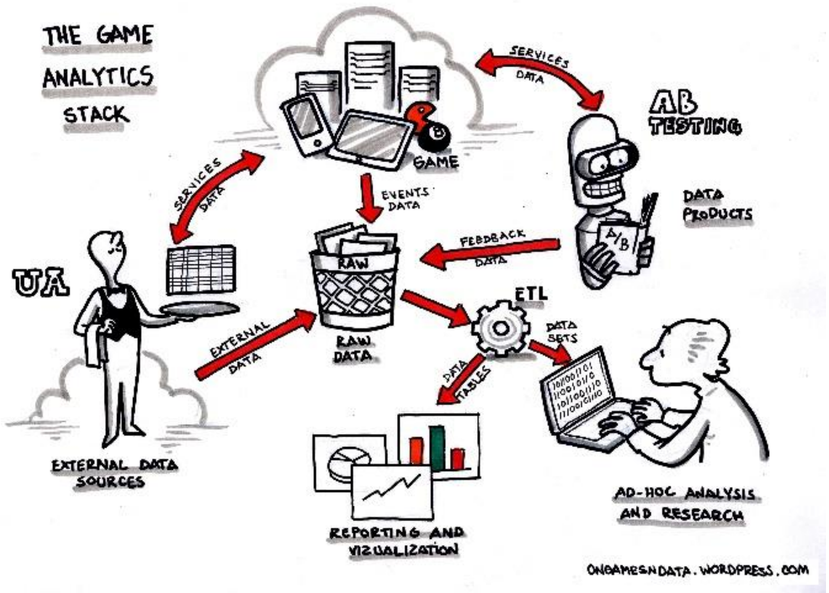
Instituto de computação uff

MEDIALAB

ESI Laboratório de Engenharia de Sistemas e Informação

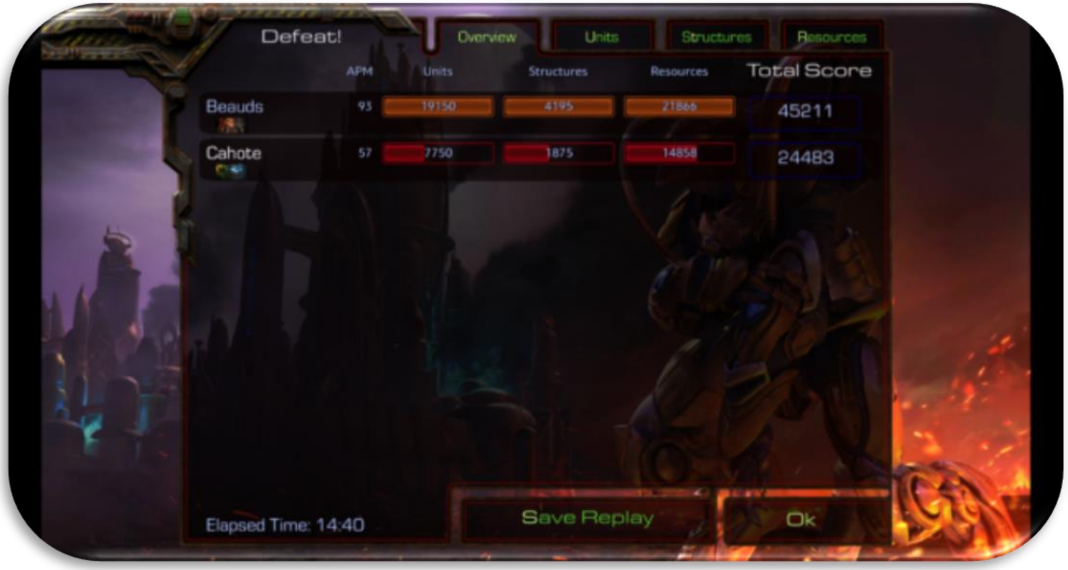
Game Analytics

- Game analytics became an emerging field
  - Important for business intelligence
  - Wealth of information
  - Designers and players
- Provides feedback
  - Design, Gameplay, behavior, experience, monetization
- Goal is to support the decision-making process for game development
  - Operational, tactical, and strategic levels



Game Analytics

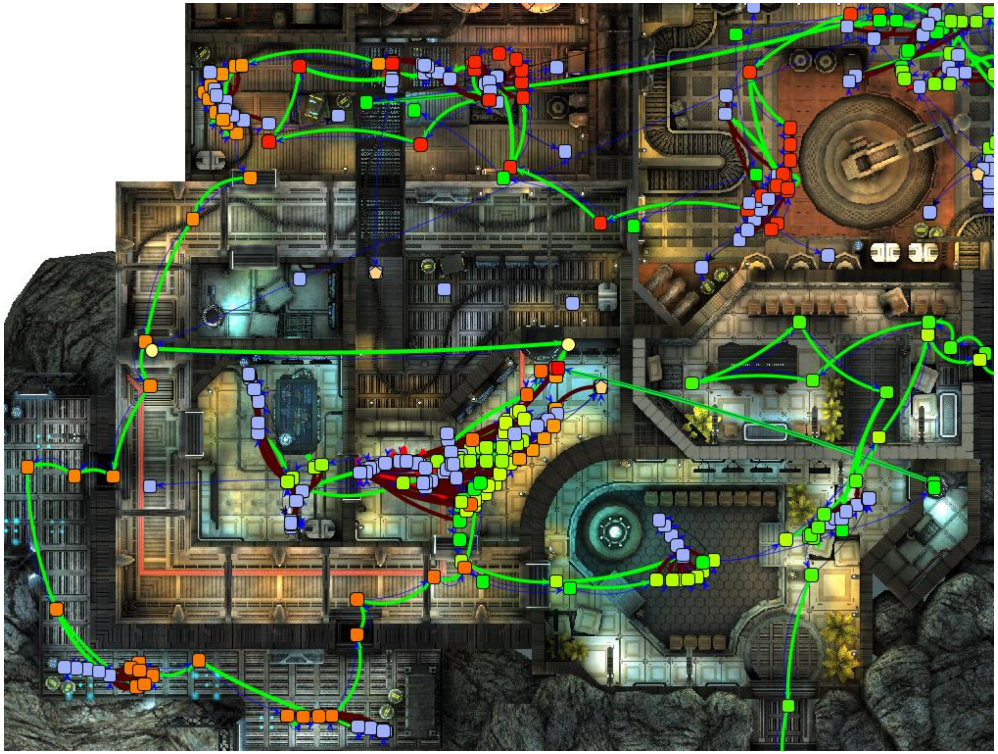
- Generates lots of data!
  - How can we analyze it to extract the knowledge that we seek?
- Player lost
  - How do we find out what went wrong?
  - What should we do to not lose again!?



Provenance in Games

PinGU – Provenance in Games
(Kohwalter et al., 2017)
http://gems-uff.github.io/ping/

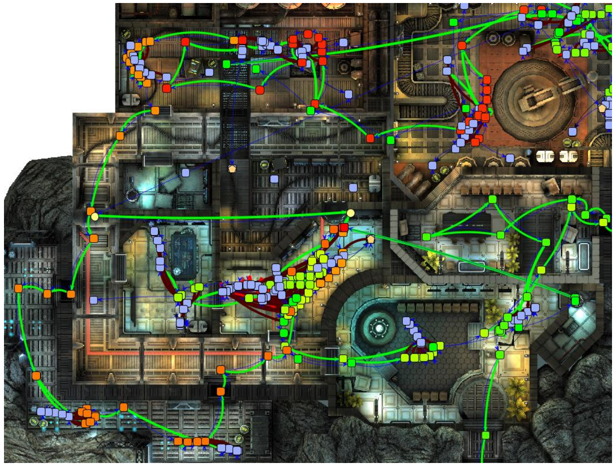
- PinG
  - SBGames 2012, 2013, **2017**, 2018, 2022
  - ACE 2013
  - SBES 2014
  - Ent Comp 2018, 2019
  - FGCS 2018
  - FoD 2020
  - ICEC 2020
- Track game session data
- Provenance graph
  - Relations between actions, agents, and entities
  - Causal relationships



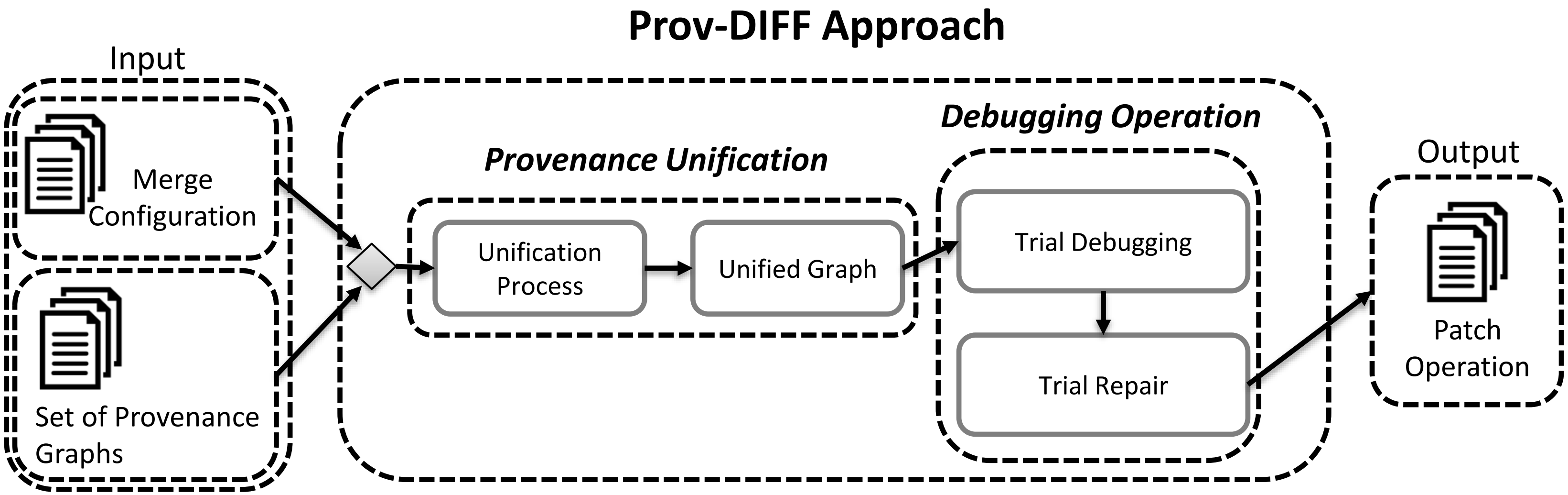
Prov-DIFF

- Propose the Prov-DIFF approach
  - – Use game session **provenance data**
  - – Understand the **reasons** for outcomes
  - – **Why** some players **failed** to achieve goals

- Debug gaming session data
  - – **Compare** players' provenance data
  - – **Detect** reasons for failing
  - – **Determine** actions to improve

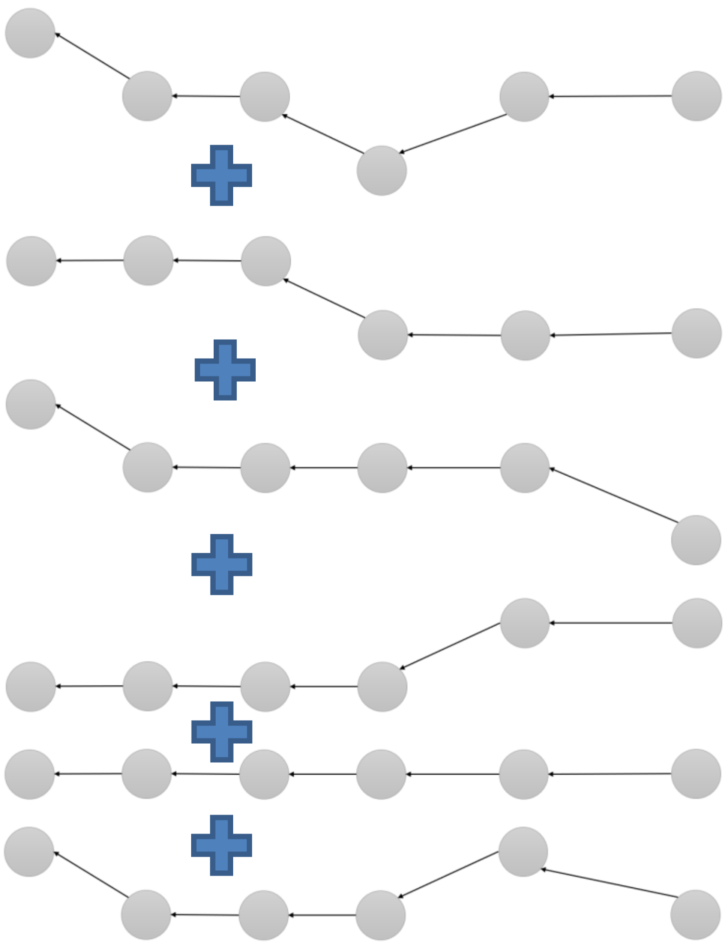


Prov-DIFF architecture



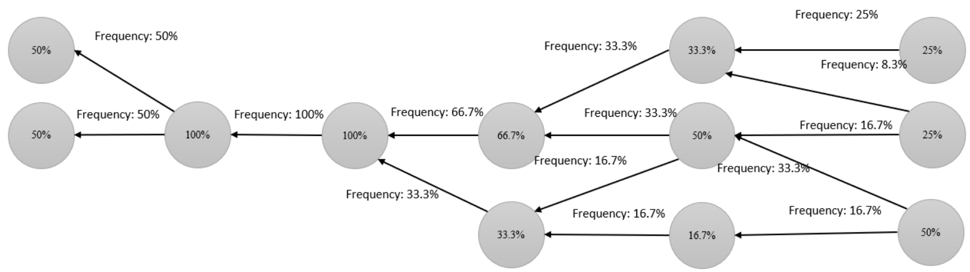
Prov Unification

N provenance graphs



...

Unified provenance graph

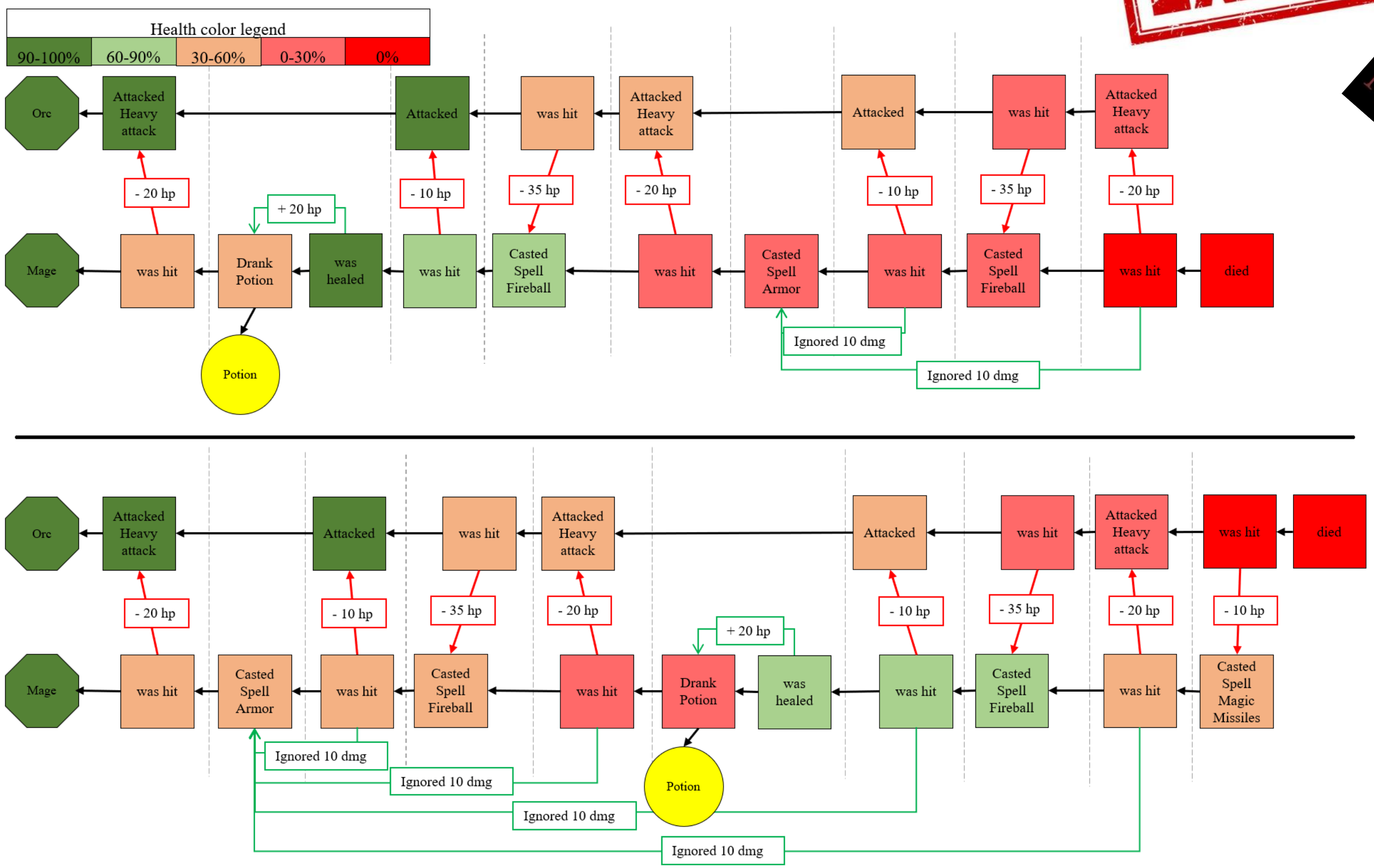


GUIDING EXAMPLE

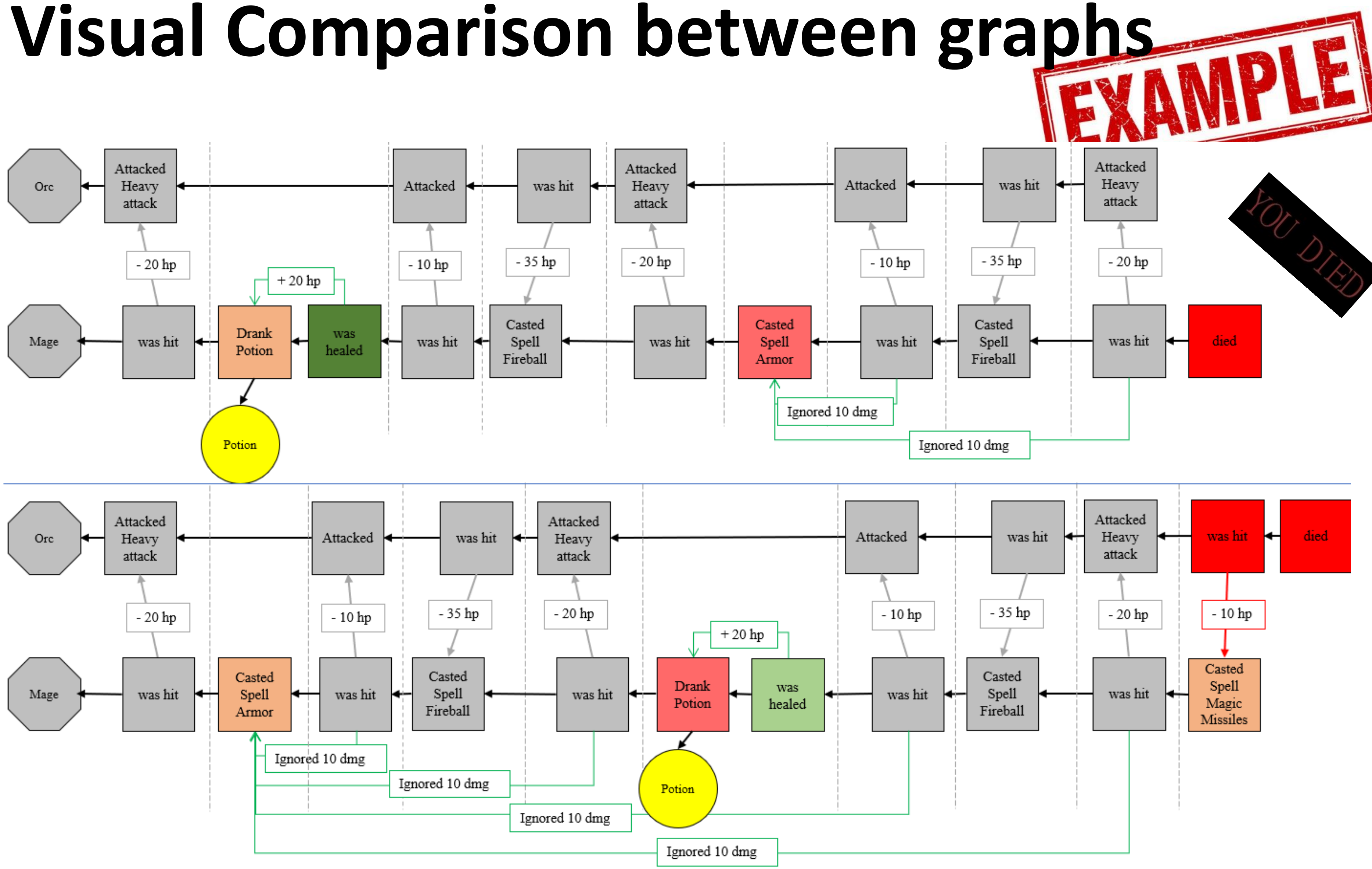
Different game sessions

EXAMPLE

YOU DIED

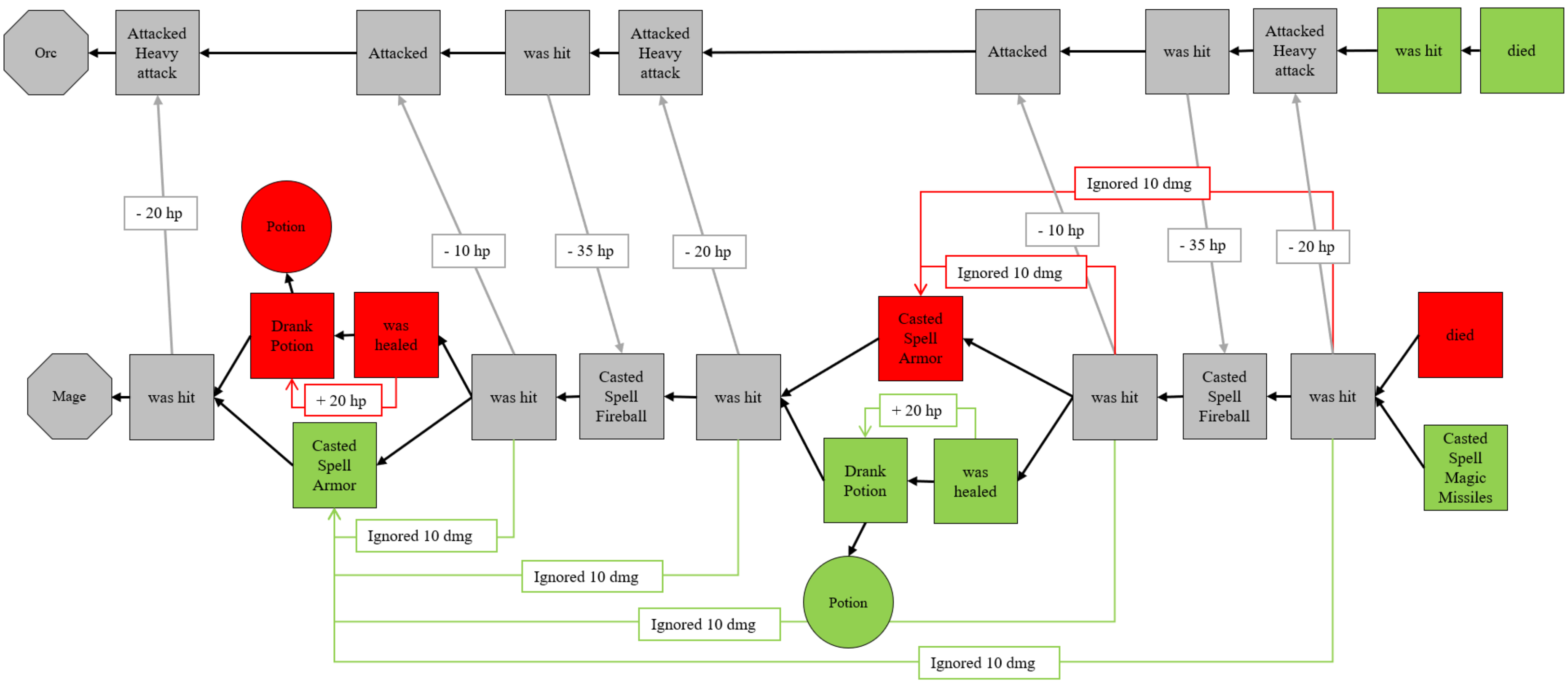


Visual Comparison between graphs



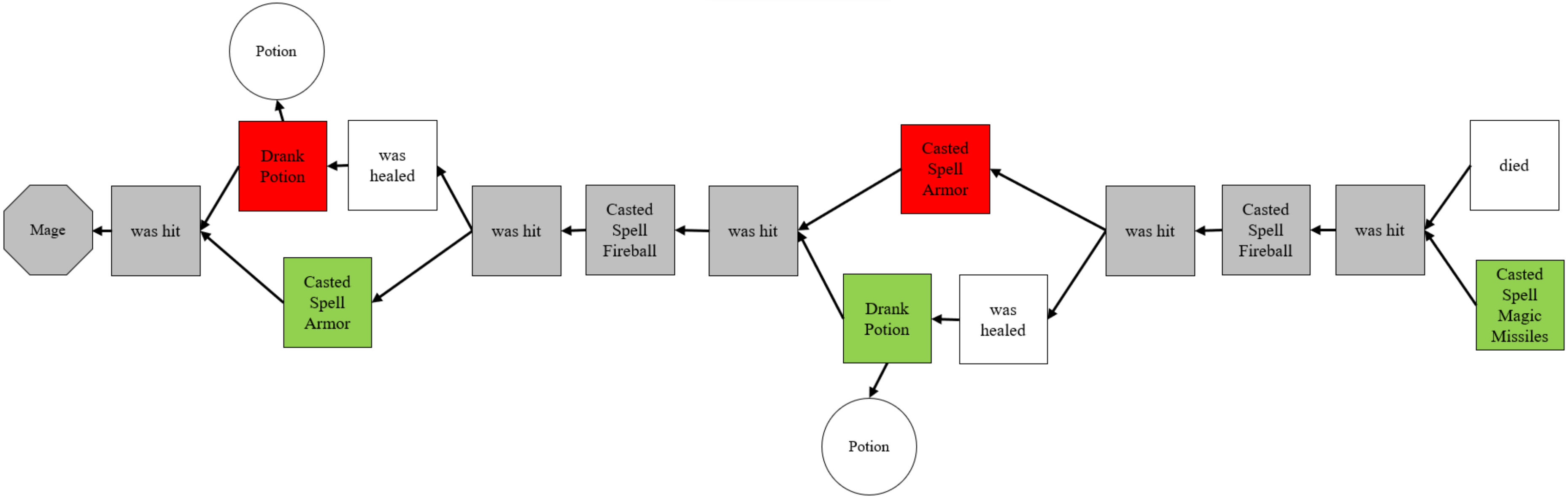
Unified Graph

EXAMPLE



Suggestion for player actions

EXAMPLE



Evaluation

***RQ1**:Can the proposed approach correctly detect the causes of the failure?*

***RQ2**: Does preserving multiple segments of the fail graph impact the results from our algorithm*?

- Three dependent variables:
  - Accuracy (correctly predicted)
  - Retention (% of preserved original actions)
  - Harmonic mean (overall performance)

Projectile Motion Simulation

- Shooting Competition
  - Nine configurable parameters
  - 15 participants
  - 20 rounds each
  - Objective: Hit the target

- Reason:
  - Controlled experiment for evaluation
  - Easy to generate data and simulate the proposed changes (re-run the simulation) for validation

Experiment Plan

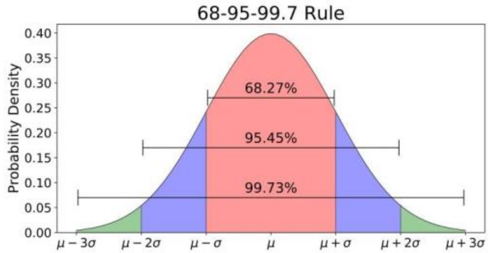
1. Generate the dataset
   - **300** generated datasets
   - Only **16** hit the target (**5.33%**)

2. Create different similarity thresholds to analyze the impact of the retention vs. accuracy
   - **10** different thresholds: varying from 0 to 3 standard deviations

3. Generate the unified graph for each similarity threshold from stage 2
   - **10** unified graphs

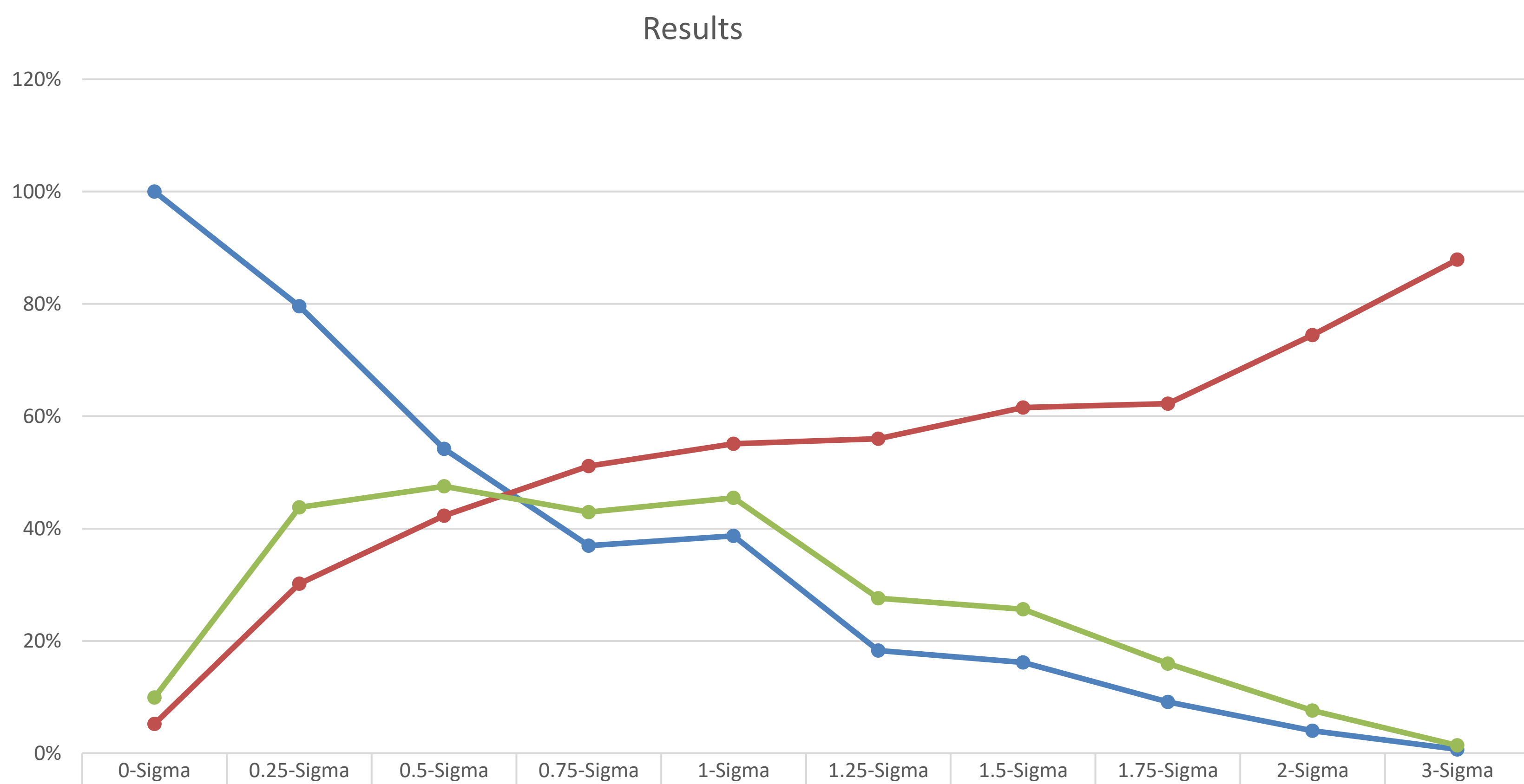
4. Execute the experiment using the unified graphs from Stage 3
   - **284** trials for each unified graph

5. Analyze the results



Evaluation Results

Accuracy = Corrected / Total_Failed
Retention = Avg % of change
Harmonic Mean = 2(Acc * Ret)/ (Acc + Ret)



	0-Sigma	0.25-Sigma	0.5-Sigma	0.75-Sigma	1-Sigma	1.25-Sigma	1.5-Sigma	1.75-Sigma	2-Sigma	3-Sigma
Accuracy	100%	80%	54%	37%	39%	18%	16%	9%	4%	1%
Retention	5%	30%	42%	51%	55%	56%	62%	62%	74%	88%
Harmonic mean	10%	44%	48%	43%	45%	28%	26%	16%	8%	1%

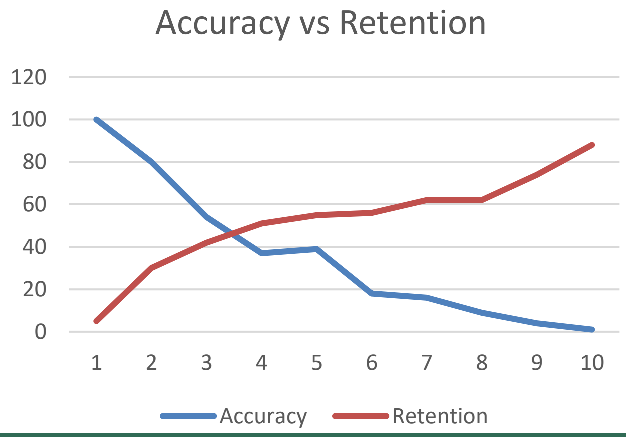
Analysis of the Results

RQ1:Can the proposed approach correctly detect the causes of the failure?

Answer: Yes, our algorithm can reach 80% accuracy when using 0.25-sigma.

RQ2: Does preserving multiple segments of the fail graph impact the results from our algorithm?

Answer: Yes, it does. The results show that the *accuracy* and *retention* metrics are inversely proportional. *Accuracy* decreases as the *retention* rate increases.
The more we preserve the original actions, the harder it is to change the outcome.



Conclusion

- Contributions:
  - Approach to determine reasons for failure
    - And suggest improvements!
    - Like spectra-based fault localization debugging…
    - Contrast players' performance with others

  - Provenance Diff

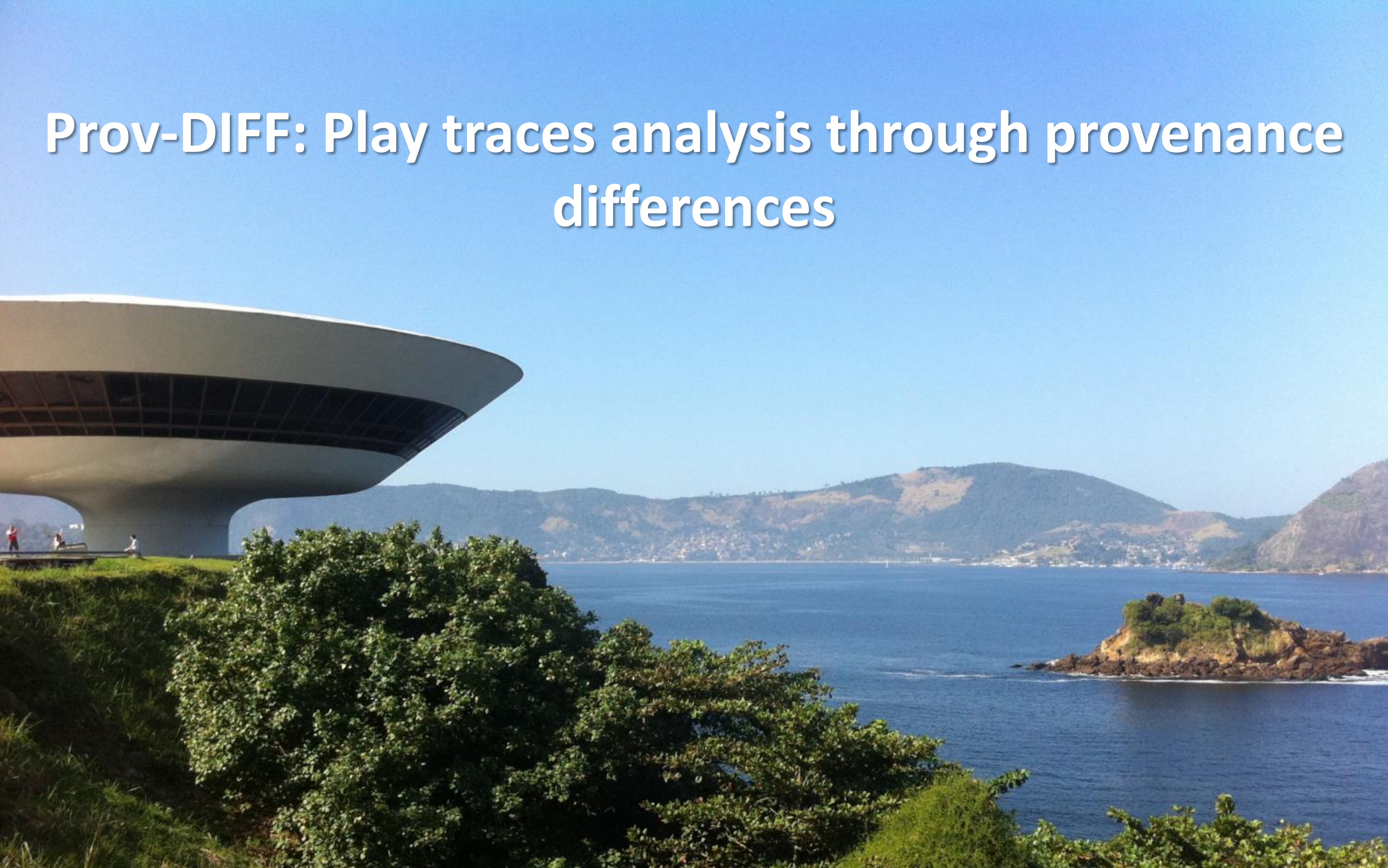
- Limitation:

  - Requires at least 1 success case

Future Work

- Integrate with **Provchastic** and **Prov-Replay**
- Find good patterns from success graphs
  - Improve success chances
- Similarly, bad patterns to avoid!
- Real game tests

Prov-DIFF: Play traces analysis through provenance differences



Authors: **Troy C. Kohwalter,** Leonardo G. P. Murta, Esteban W. G. Clua

{troy, leomurta, esteban}@ic.uff.br

Instituto de Computação – Universidade Federal Fluminense (UFF)

Instituto de computação

MEDIALAB

Laboratório de Engenharia de Sistemas e Informação

Unified Graphs

Similarity Threshold	# Total Vertices	# Only Success Vertices	# Only Fail Vertices	# Common Vertices
0.00-Sigma	2420	128	2282	10
0.25-Sigma	142	0	78	64
0.50-Sigma	87	0	38	49
0.75-Sigma	66	0	27	39
1.00-Sigma	58	0	21	37
1.25-Sigma	54	0	17	37
1.50-Sigma	53	0	18	35
1.75-Sigma	50	0	11	39
2.00-Sigma	40	0	13	27
3.00-Sigma	18	0	2	16