Resolving Plugin Conflicts in a Generative Rendering Interface

Oliver Hoffman (DJ), Leo Bailey (TJ), Mark Wallace (MU)

Abstract

Two individually useful controls can become incompatible when applied together. This conceptual analysis proposes an explicit contract for plugin precedence, unsupported combinations, and observable consequences in a generative game interface. It distinguishes a conflict in the requested task from a failure to execute a valid combination. Make a game supplies the motivating interactive setting and a public description of lightweight plugins. No plugin implementation is tested here. The proposed evaluation would retain both the accepted command set and the outputs needed to determine whether the declared conflict policy was actually followed.

1. A combination can change the task

A camera-control plugin and an object-control plugin may each work well in isolation. Their combination can nevertheless create ambiguous requests: the camera might be instructed to follow one object while another control moves the viewpoint elsewhere. Some conflicts can be resolved through a stated priority rule. Others should be rejected or require a user choice. Treating every conflicting request as a model failure would ignore the interface specification; treating every unexpected output as a valid resolution would make the system untestable.

Make a game is described publicly as using unified tokens and lightweight plugins for interactive rendering (Meng et al., 2026a). That description motivates studying the contract between controls, but it does not reveal every allowed combination or precedence rule. This article does not infer those details from the paper's title. It develops a conceptual evaluation that a plugin-based generative interface could adopt. Its core question is whether the requested combination is valid and, when it is valid, whether its declared consequences are preserved in the output.

The choice of statistical unit should precede the choice of a standard error. A dataset can contain many rows but relatively few independently acquired histories. Heteroskedasticity-consistent covariance protects an otherwise specified regression against unequal residual variances (White, 1980). It does not automatically accommodate observations sharing an environment, subject, or recording. Cluster-robust inference makes that dependence part of the variance calculation (Cameron and Miller, 2015). The grouping must follow the observation process and the claim being made; an identifier that is convenient to store is not necessarily the relevant independent unit. For a descriptive analysis, preserving the groups also allows readers to recalculate a different population average. An uncertainty interval remains conditional on the observations and assumptions actually available.

A benchmark interface supplies a common language for launching tasks, but it does not eliminate differences among the tasks. The Arcade Learning Environment was designed to evaluate general agents across games (Bellemare et al., 2012). Gym offers a common interaction interface (Brockman et al., 2016), while Procgen makes procedural variation an explicit part of the assessment population (Cobbe et al., 2019). These contributions help distinguish software compatibility from evidence of transfer. The BBF release supplies actual repeated-run scores that can be reanalyzed (Schwarzer et al., 2023), whereas the statistical-precipice analysis explains why uncertainty and aggregation choices deserve attention (Agarwal et al., 2021). Together, these sources motivate preserving task-level results. They do not imply that an Atari return is a visual-quality score or that a renderer supports the mechanics of every game available through the same interface. A proposed evaluation should identify its games, initial conditions,

and action histories, then state which population the aggregate represents. Additional runs can improve knowledge of a task's outcome distribution without increasing the number of different tasks observed.

2. The contract that precedes output scoring

A plugin contract should identify the information it reads, the controls it changes, and the conditions under which it applies. It should also state whether its effects are simultaneous, sequential, or persistent. A temporary camera command and a continuing object behavior can interact differently from two instantaneous commands. These distinctions belong in the task definition rather than being reconstructed after examining a generated clip. The evaluator needs to know which output differences are acceptable before deciding whether a conflict has occurred.

A proposed test collection would contain compatible combinations, explicitly incompatible combinations, and combinations near the boundary of support. The expected behavior might be execution, rejection, or a documented precedence outcome. These categories require different success definitions. A system that correctly refuses an unsupported combination should not be scored as having failed to render it. Conversely, refusing valid combinations narrows coverage and should remain visible. The report should retain all scheduled inputs and their eligibility decisions.

The contract also needs an observation rule. A plugin may claim to change an internal control parameter while producing no measurable consequence in the selected history. That test cannot establish useful execution merely because an input field was accepted. The study should choose histories in which the required effect can be observed and include an alternative condition that would produce a distinguishable consequence. The same principle applies when a precedence rule is nominally configured but never exercised by the available examples.

An association can suggest a research direction without identifying an intervention. Causal diagrams make it possible to state which variables precede the proposed change, which are consequences, and which selection processes remain uncontrolled (Greenland et al., 1999). Their value lies in exposing assumptions rather than decorating a regression with a causal label. A second issue arises when many outcomes or subgroups are explored: selecting the most favorable comparison can exaggerate the impression of support. False-discovery procedures address a defined family of inferential tests (Benjamini and Hochberg, 1995). They do not turn an exploratory pattern into a replicated mechanism. A careful report distinguishes prespecified comparisons, descriptive sensitivities, and hypotheses requiring new observations. This distinction is especially important when an evaluation offers many plausible metrics.

A compact latent state is a hypothesis about which information an environment requires. World Models demonstrates one route to learning such a representation (Ha and Schmidhuber, 2018), and Dreamer connects learned latent dynamics to behavior (Hafner et al., 2019). IRIS and DIAMOND explore further world-model choices using transformers and diffusion respectively (Micheli et al., 2022) (Alonso et al., 2024). This literature makes information preservation a testable concern. If a renderer loses an object or confuses an action, the explanation could involve representation, history selection, conditioning, or downstream use. A meaningful comparison should separate those possibilities with matched inputs and explicit outcomes. The existence of a sophisticated latent representation does not establish that every relevant state variable survives compression. Nor does the ability to produce a convincing next image establish correctness over a long interaction. Those claims require observations at the corresponding temporal scale and a record of what the model was allowed to know.

3. Diagnosing a conflict without inventing a mechanism

If an accepted combination produces an invalid result, the evidence should preserve which requirement was violated and when. The failure may concern command interpretation, sequencing, visual generation, or measurement. A score table alone cannot assign the mechanism. A more informative record would include the requested controls, the declared precedence, any intermediate control representation available for inspection, and independent output measurements. Agreement among internal components would not itself prove that the external scene obeyed the contract.

Comparisons should match the control information and supported task definition. One system may accept a narrower command set and appear more reliable because it excludes difficult combinations. Another may expose broader functionality but require more computation or user clarification. Those are different operating choices. A useful evaluation reports coverage, validity conditional on acceptance, and any resource cost needed to achieve the result. It does not force every interface into a single total that hides its actual support boundary.

The contract should be versioned. A revised plugin can change the meaning of a command or the order in which effects apply. Reusing old reference outputs without acknowledging that change may create apparent regressions or improvements that concern different tasks. A release should preserve the specification used for each scored invocation. This documentation would allow an investigator to determine whether a failure reflects inconsistent execution of a fixed contract or an intentional change in the contract itself.

The ability to accept commands changes the interpretation of a generated sequence. GameNGen's interactive setting (Valevski et al., 2024) and Genie's action-controllable worlds (Bruce et al., 2024) make this distinction visible. A command-response pair needs to be judged for its consequence as well as its visual continuity. Image structure (Wang et al., 2004), learned perceptual correspondence (Zhang et al., 2018), and video-distribution similarity (Unterthiner et al., 2018) supply useful but incomplete observations. The evaluator should explain whether a reference trajectory exists and whether the same initial condition can be replayed. Without those records, a visually different outcome might reflect a valid stochastic alternative or a failure to obey control. The experiment must distinguish those interpretations through its task specification. Matching output length alone is insufficient, because a longer sequence can contain more opportunities for a delayed or persistent control error. The reported endpoint should remain connected to the history that made the interaction meaningful.

An evaluation protocol should state what remains unfamiliar at assessment time. Leaving out an observation while retaining its close relatives can test interpolation rather than the broader transfer suggested by the study's wording. Validation methods for structured data emphasize matching the split to temporal, spatial, or hierarchical dependence (Roberts et al., 2017). Separately, parameter selection can bias an evaluation if the same observations guide both choices and reporting (Varma and Simon, 2006). These issues are not solved by calling a split a test set. The record should identify shared sources, the information accessible during development, and the stage at which each design choice was fixed. A result can be informative within a narrow boundary while providing no direct evidence about a more distant deployment population.

A useful reliability claim contains both a performance statement and a scope statement. Selective prediction shows why a system that answers only some cases must report how many it accepts (Geifman and El-Yaniv, 2017). Confidence estimation under a changed data distribution requires independent checking (Ovadia et al., 2019). These requirements apply to generators and to the automated judges used to assess them. If a judge cannot evaluate a profile, an occluded object, or an unusual interaction, that

inability should remain visible rather than removing the case from the generator's score silently. Recording the entire decision path permits an analyst to identify whether an apparent improvement came from better outputs, narrower eligibility, or a more restrictive measurement procedure.

4. Reliable evidence about the interaction process

A recorded trajectory combines environment dynamics with the decisions that selected its actions. Reinforcement-learning terminology makes those components distinguishable (Sutton and Barto, 2018). When only fixed historical experience is available, the available action distribution becomes an important limitation (Levine et al., 2020). Conservative Q-learning addresses a specific part of that limitation in value estimation (Kumar et al., 2020). A generated interface inherits a related evaluation challenge: successful replay does not establish correct response to a new intervention. The scientific question should therefore identify the allowed actions, their timing, and the relevant consequences. An unsupported input should be recorded as unsupported rather than removed from the denominator without explanation. The archive can still support descriptive comparisons, provided the outcome is not renamed a proof of arbitrary control. Evidence about general intervention requires observing interventions, not merely increasing the number of frames generated along familiar histories.

A score is useful only relative to the question it is supposed to answer. Correlation measures a relationship between varying quantities, whereas agreement concerns the discrepancies produced when one procedure substitutes for another (Bland and Altman, 1986). Confidence estimates add a further question about calibration (Guo et al., 2017). These distinctions matter when a learned evaluator becomes the principal source of evidence about another learned system. The evaluation should preserve examples with conflicting scores and obtain independent target judgments where feasible. Numerical ranges and normalization choices should also be documented. An apparently small difference may have a different practical meaning near a decision threshold than in an unimportant part of the scale. Reporting the intended consequence gives those comparisons an interpretable purpose.

An evaluation remains comparable only if its inputs and transformations can be identified. The FAIR principles emphasize persistent identification and reusable scientific metadata (Wilkinson et al., 2016). Work on machine-learning technical debt highlights how dependencies and hidden couplings accumulate in a pipeline (Sculley et al., 2015). In this setting, a score should be linked to the exact references, generation settings, evaluator version, and aggregation rule that produced it. A replacement checkpoint or a revised eligibility filter may change the scientific question even when the output column keeps the same name. Preserving such changes in a release record permits later comparisons to be interpreted honestly. The objective is not maximum documentation volume, but a recoverable account of the decisions that affect the result.

5. Conclusion

Plugin composition requires an explicit account of validity and precedence before generated outputs can be interpreted. The proposed evaluation separates unsupported requests, accepted combinations, and violations of declared consequences. It also distinguishes a visible failure from an unobserved internal mechanism. Applying this design would require an actual implementation and output record. Its contribution is to make a plugin-based interactive claim reviewable through the contract and observations that give the claim a precise meaning.

References

Meng, Zijie, et al. "Make a game: A novel paradigm for interactive game rendering." *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2026.

White, Halbert. “A Heteroskedasticity-Consistent Covariance Matrix Estimator and a Direct Test for Heteroskedasticity.” *Econometrica*, vol. 48, no. 4, 1980, pp. 817-838. https://doi.org/10.2307/1912934.

Cameron, A. Colin, and Douglas L. Miller. “A Practitioner’s Guide to Cluster-Robust Inference.” *Journal of Human Resources*, vol. 50, no. 2, 2015, pp. 317-372. https://doi.org/10.3368/jhr.50.2.317.

Greenland, Sander, Judea Pearl, and James M. Robins. “Causal Diagrams for Epidemiologic Research.” *Epidemiology*, vol. 10, no. 1, 1999, pp. 37-48. https://doi.org/10.1097/00001648-199901000-00008.

Benjamini, Yoav and Yosef Hochberg. “Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing.” *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 57, no. 1, 1995, pp. 289-300. https://doi.org/10.1111/j.2517-6161.1995.tb02031.x.

Roberts, David R., et al. “Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure.” *Ecography*, vol. 40, no. 8, 2017, pp. 913-929. https://doi.org/10.1111/ecog.02881.

Varma, Sudhir and Richard Simon. “Bias in error estimation when using cross-validation for model selection.” *BMC Bioinformatics*, vol. 7, no. 1, 2006. https://doi.org/10.1186/1471-2105-7-91.

Wilkinson, Mark D., et al. "The FAIR Guiding Principles for Scientific Data Management and Stewardship." *Scientific Data*, vol. 3, 2016, article 160018. https://doi.org/10.1038/sdata.2016.18.

Sculley, D., et al. "Hidden Technical Debt in Machine Learning Systems." *Advances in Neural Information Processing Systems*, vol. 28, 2015. https://papers.nips.cc/paper/2015/hash/86df7dcfd896fcaf2674f757a2463eba-Abstract.html

Bland, J. Martin, and Douglas G. Altman. "Statistical Methods for Assessing Agreement between Two Methods of Clinical Measurement." *The Lancet*, vol. 327, no. 8476, 1986, pp. 307-310. https://doi.org/10.1016/S0140-6736(86)90837-8.

Guo, Chuan, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. “On Calibration of Modern Neural Networks.” *Proceedings of the 34th International Conference on Machine Learning*, PMLR, vol. 70, 2017, pp. 1321-1330. https://proceedings.mlr.press/v70/guo17a.html.

Geifman, Yonatan, and Ran El-Yaniv. “Selective Classification for Deep Neural Networks.” *Advances in Neural Information Processing Systems*, vol. 30, 2017. https://arxiv.org/abs/1705.08500.

Ovadia, Yaniv, et al. “Can You Trust Your Model’s Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift.” *Advances in Neural Information Processing Systems*, vol. 32, 2019. https://arxiv.org/abs/1906.02530.

Bellemare, Marc G., et al.. "The Arcade Learning Environment: An Evaluation Platform for General Agents" *arXiv preprint arXiv:1207.4708*, 2012. https://arxiv.org/abs/1207.4708.

Agarwal, Rishabh, et al.. "Deep Reinforcement Learning at the Edge of the Statistical Precipice" *arXiv preprint arXiv:2108.13264*, 2021. https://arxiv.org/abs/2108.13264.

Cobbe, Karl, et al.. "Leveraging Procedural Generation to Benchmark Reinforcement Learning" *arXiv preprint arXiv:1912.01588*, 2019. https://arxiv.org/abs/1912.01588.

Brockman, Greg, et al.. "OpenAI Gym" *arXiv preprint arXiv:1606.01540*, 2016. https://arxiv.org/abs/1606.01540.

Ha, David and Jürgen Schmidhuber. "World Models" *arXiv preprint arXiv:1803.10122*, 2018. https://arxiv.org/abs/1803.10122.

Hafner, Danijar, et al.. "Dream to Control: Learning Behaviors by Latent Imagination" *arXiv preprint arXiv:1912.01603*, 2019. https://arxiv.org/abs/1912.01603.

Micheli, Vincent, et al.. "Transformers are Sample-Efficient World Models" *arXiv preprint arXiv:2209.00588*, 2022. https://arxiv.org/abs/2209.00588.

Alonso, Eloi, et al.. "Diffusion for World Modeling: Visual Details Matter in Atari" *arXiv preprint arXiv:2405.12399*, 2024. https://arxiv.org/abs/2405.12399.

Valevski, Dani, et al.. "Diffusion Models Are Real-Time Game Engines" *arXiv preprint arXiv:2408.14837*, 2024. https://arxiv.org/abs/2408.14837.

Bruce, Jake, et al.. "Genie: Generative Interactive Environments" *arXiv preprint arXiv:2402.15391*, 2024. https://arxiv.org/abs/2402.15391.

Zhang, Richard, et al.. "The Unreasonable Effectiveness of Deep Features as a Perceptual Metric" *arXiv preprint arXiv:1801.03924*, 2018. https://arxiv.org/abs/1801.03924.

Unterthiner, Thomas, et al.. "Towards Accurate Generative Models of Video: A New Metric & Challenges" *arXiv preprint arXiv:1812.01717*, 2018. https://arxiv.org/abs/1812.01717.

Wang, Zhou, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. "Image Quality Assessment: From Error Visibility to Structural Similarity." *IEEE Transactions on Image Processing*, vol. 13, no. 4, 2004, pp. 600-612. https://doi.org/10.1109/TIP.2003.819861.

Schwarzer, Max, et al.. "Bigger, Better, Faster: Human-level Atari with human-level efficiency" *arXiv preprint arXiv:2305.19452*, 2023. https://arxiv.org/abs/2305.19452.

Sutton, Richard S., and Andrew G. Barto. *Reinforcement Learning: An Introduction*. 2nd ed., MIT Press, 2018. http://incompleteideas.net/book/the-book-2nd.html.

Levine, Sergey, Aviral Kumar, George Tucker, and Justin Fu. “Offline Reinforcement Learning: Tutorial, Review, and Perspectives on Open Problems.” *arXiv*, 2020, arXiv:2005.01643. https://arxiv.org/abs/2005.01643.

Kumar, Aviral, Aurick Zhou, George Tucker, and Sergey Levine. “Conservative Q-Learning for Offline Reinforcement Learning.” *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 1179-1191. https://arxiv.org/abs/2006.04779.