Supplementary Figure 1: HA and MP segment phylogenies (partial) for isolates from Holmes et al. The set of strains in the phylogenetic neighbourhood of A/New York/182/2000 seem to be consistent between the two segment phylogenies suggesting that it is unlikely to be a reassortant (consensus trees from GiRaF, drawn using the program FigTree, http://tree.bio.ed.ac.uk/software/figtree/).
Supplementary Figure 2: Size distribution of implanted and candidate reassortments. The graph shows the frequency of reassortants as a function of the size of true positive, false positive and implanted reassortments for the “All Events” experiment described in Table 1. Interestingly, while the size distribution of true positives matches that of the implants, false positives are more often large sets (≥ 6 taxa) and hence amenable to manual filtering. The trend of false positives being large sets is also more pronounced with multiple reassortments (data not shown).

Supplementary Figure 3: Confidence values for candidate reassortments. a) Tradeoffs using various confidence value thresholds for the “All Events” experiment described in Table 1. b) Calibration of confidence values for the same experiment – in the ideal case the histogram shown would have a single peak (for the range 0-0.005) indicating that false-discovery-rate = 1 – confidence-value in all cases.
Supplementary Figure 4: Schematic for computational steps in GiRaF.

Supplementary Figure 5: Reassortment events in human influenza A (H3N2) isolates. Detailed consensus trees, in following pages, for a) HA segment and b) NA segment for the corresponding condensed trees in Figure 1.

Supplementary Table 1: Catalog of reassortant H5N1 Strains. The table (additional excel file) contains a list of 18 candidate reassortments and their reassortment architectures that were automatically identified by GiRaF based on all pairwise segment comparisons for a set of H5N1 genomes.

Supplementary Table 2: Catalog of reassortant Swine Influenza & S-OIV strains. The table (additional excel file) contains a list of 37 candidate reassortments among swine influenza strains as identified by GiRaF through a fully automated analysis. Representatives of the S-OIV strains involved in the recent “swine flu” outbreak were also included in the dataset and identified correctly as reassortants by GiRaF.
a) HA segment
b) NA segment
Alternative Tree Topology

To investigate the effect of tree topology on the results from our simulated datasets, we also generated datasets using an alternative topology. Specifically, we used a neighbor-joining tree for the HA segment of isolates from the S-OIV and swine influenza sequences and repeated the other steps of our experiment as described in the text. While the number of taxa in this set is similar to that in the original set (140 as compared to 156), the topology is more clearly divergent. In general, we obtained similarly good results using GiRaF on these datasets (for the “All Events” set, sensitivity was 94% and PPV was 90%) suggesting that the results reported in the text are a reasonable measure of GiRaF’s performance in general.

Performance of Distance Test in Isolation

To assess the utility of a distance test independent of information from tree topology in predicting reassortments we experimented with a heuristic approach based on Rabdan et al. For this, we used the test described in Rabdan et al. (only steps 1 & 2) to compare all pairs of taxa and identify those that have diverged with respect to each other (bonferroni-corrected E-value threshold of 0.01). Taxa that had identical profiles of divergence were then clustered into putative reassortments and sets of size less than 20 were reported. In general, this approach performed poorly and compared to GiRaF on the “All Events” set had low sensitivity (35%) and PPV (20%) values.

Identifying Potential Parents

For reassortments identified by GiRaF, potential parents in each segment can be computed using the script “get_parents.pl” that is provided with it. This script scans the set of splits for each segment and identifies those splits that contain the reassortment set entirely on one side. The corresponding sets (that contain the reassortment set) are considered potential parents with a confidence value given by the split. The script then selects the smallest sets with confidence value greater than 0.5 (the least common ancestors) and among these the most confident set is reported as the potential parent in each segment.

Biclique Finding

The biclique-finding algorithm used in GiRaF is based on the idea of using the consensus operation to enumerate through the space of high-confidence bicliques in the incompatibility graph for two segments (say A and B). Given two bicliques (L_1, R_1) and (L_2, R_2) (where L_i and R_i are the vertex sets on the “left” and “right” of the biclique), the consensus of the bicliques is given by (L_1 \cup L_2, R_1 \cap R_2). Given two maximal bicliques (vertex sets are not contained in another bilcique), the consensus operation can be shown to always produce another maximal biclique. The algorithm used in GiRaF then uses the following steps to enumerate all maximal bicliques (L_i, R_i) which have high confidence (i.e. C_A(L_i) > T and C_B(R_i) > T, where C_X(Y) is the confidence value associated with a set of splits Y based on trees for segment X and T is a confidence threshold parameter, that is set at 0.7 in GiRaF):

1. Let S be the set of all star bicliques (L, R_i) (|L|=1 and R_i contains all neighbors of the node in L) s.t. C_B(R_i) > T.
2. Set F to S
3. For each biclique \( B_i \) in \( S \) and \( B_j \) in \( F \), let \( D = (L, R) \) be the consensus of \( B_i \) and \( B_j \). If \( C_B(R) > T \), add \( D \) to \( F \).
4. If new sets were added to \( F \) in 3, repeat 3, considering only the new additions in \( F \).
5. Filter out all bicliques \((L_i, R_i)\) in \( F \) where \( C_A(L_i) \leq T \). Report \( F \) as the set of all high-confidence maximal bicliques in the graph.

While the runtime of this algorithm is linear in the number of maximal bicliques (which can be exponential in the size of the graph), in practice, the restriction to high-confidence bicliques reduces the search space drastically and makes the runtime feasible even for large graphs.