Supplementary Methods

Generation of Illumina paired-end sequence data

Strains. We obtained one female mouse each from the C57BL/6J (B6) and DBA/2J (DBA) inbred strains (Jackson Laboratory, Bar Harbor, ME). The B6 individual was obtained in January, 2006. This mouse was a retired foundation stock breeder (F226), and is thus derived from the colony nucleus and should be minimally diverged from the reference genome (Waterston et al. 2002). The DBA mouse was obtained in December, 2004. This mouse was ordered through the standard mechanism and is not pedigreed. These are precisely the same samples analyzed in our previous aCGH study (Egan et al. 2007).

DNA preparation. DNA was isolated using the Puregene Kit (Gentra Systems), with the following modifications. Livers were flash frozen in liquid nitrogen, ground with mortar and pestle, and dounce-homogenized in 20 ml cell lysis solution. After isolation of crude DNA from liver and tail, further extractions were performed with phenol/chloroform/isoamyl alcohol, and chloroform/isoamyl alcohol, and DNA was precipitated with 2 volumes isopropanol and 1/10 volume sodium acetate.

Paired-end sequencing. We constructed paired-end sequencing libraries according to the manufacturers protocols, as described (Bentley et al. 2008). For each strain we constructed 5-8 independent libraries. Libraries were sequenced on the Illumina GA2 housed in the University of Virginia School of Medicine core facility. Read lengths ranged from 31-44bp with a mean of 39bp (Table 1).

Paired-end sequence alignment and classification

Removal of low-quality and low-complexity paired-end reads. Prior to aligning paired-end Illumina reads (matepairs), we first excluded all matepairs that did not pass Illumina's quality threshold. We further enforced that all remaining matepairs have no more than three Ns on either end. Lastly, we required that on both ends of the matepair, a single base (including Ns) does not comprise more than 80% of the sequence. These restrictions resulted in a total of 130.2 and 74.7 million DBA and B6 matepairs, respectively (see Fig. S3 for further details).

Sequence alignment with BWA. We aligned all matepairs that passed our quality and complexity filters with the BWA (Li and Durbin 2009) (version 0.4.4) alignment algorithm. We found that BWA provides reasonable alignment sensitivity while using minimal computer memory and disk space. We use BWA as a preliminary screen for matepairs that are concordant with the mm9 reference sequence. BWA

aligns each end of each pair separately and then uses a "pairing" process to find concordant matepairs among the alignments for each end. We sought to find a substantial fraction of the concordant matepairs with BWA and, as such, we used rather sensitive alignment parameters to the detriment of alignment speed.

During the alignment phase, we used an alignment seed size of 20 ("-l 20"), and allowed for up to two differences within the seed ("-k 2") and up to 8 differences in each end of each read ("-n 8"). We permitted up to 3 gaps to be opened in the alignment ("-o 3") and up to 3 gap extensions ("-e 3"). We also forced BWA to continue searching for suboptimal alignments, even in cases where the best alignment was to a repetitive sequence ("-R").

In the alignment pairing phase, we specified that the maximal expected insert size ("-a") should be equivalent to the median fragment size plus 10 times the median absolute deviation of the DNA fragment library. In contrast to measures based on standard deviation, this measure of DNA fragment variation is less susceptible to gross over-estimation of variability owing to large outliers. We also allowed up to 10 million possible mapping locations ("-o 10000000") for each end in order to prevent missing concordant matepairs for highly-repetitive sequences.

After the alignments were paired by BWA, we set aside all concordant matepairs for CNV detection via depth of coverage (DOC) analyses (see below). BWA is amenable to such analyses as it chooses a random mapping location for matepairs that map concordantly to multiple locations in the genome, thus minimizing systematic coverage distribution biases.

Sequence alignment with NOVOALIGN. All remaining matepairs that were either discordant with or did not align to the reference genome were subsequently re-aligned with NOVOALIGN (C. Hercus, unpublished: http://www.novocraft.com/products.html#novoalign). We found NOVOALIGN (version 1.05.01) to be a more sensitive aligner (data not shown) and therefore used it as a secondary screen for additional concordant matepairs that were missed by BWA. We separately aligned each end of each remaining pair with sensitive settings (word size of 14, step size of 1, -g 0, -x 30, -r E, -t 90, -e 5000000) and recorded all possible mapping locations for each end of each pair. Using our custom software (REFMAPPER), we paired the NOVOALIGN alignments for each end of each pair and screened for combinations that proved to be concordant with the reference genome. For those matepairs that were still found to be discordant, we computed all possible mapping combinations (e.g. if end 1 and end 2 each have 10 mappings, there are 100 total mapping combinations). The discordant mapping combinations for all discordant matepairs were then screened for sequencing artifacts, fragment redundancy, and low-complexity sequence (e.g. SSRs) (see below). All remaining discordant mappings were used to find structural variations with HYDRA.

Alignment of matepairs forming SVs with MEGABLAST. As a final means to eliminate false positive SV calls that arose because both BWA and NOVOALIGN failed to find matepairs that were in fact concordant, we used MEGABLAST (Zhang et al. 2000) to re-align all putatively-discordant matepairs that comprised HYDRA SV calls. We found MEGABLAST to be the only aligner tested to have the required sensitivity and speed to find concordant mappings for putative SV calls in a reasonable timeframe. We aligned (word size of 8, -G 8, -E 2, -a 6, -F F, -q -2, -r 2, -D 3) each discordant read-pair from each putative SV call with MEGABLAST and asked if any of the matepairs in an SV call were found to be concordant. If so, we classified the SV call as a low-confidence variant owing to the possibility that it was observed merely because of a lack of alignment sensitivity. SV calls where no matepairs were found to be concordant were classified as high-confidence variants.

Removal of sequencing artifacts and low complexity sequence.

Removal of redundant sequence fragments. In order to minimize false positive SV calls, we sought to exclude all "redundant" matepairs that arise from sequencing the same molecule more than once. Otherwise, redundant matepairs would be falsely interpreted as independent measurements and would lead to false positive variant calls. We examined all mappings for all of the discordant matepairs and searched for two or more matepairs that shared the same alignment start and end coordinates (+/- 2bp). In such cases, we retained the matepair with the least edit distance relative to the reference genome and excluded all of the mappings from the other matepairs from our analysis. Interestingly, we observed that the majority of redundant matepairs can be traced to nearby coordinates on the Illumina flow-cell. This suggests that the source of most redundancy is independent base-calling of sub-clusters formed during the bridge PCR step on the flowcell. We have found that this effect is inversely proportional to cluster density on the flow-cell and can be partially mitigated by loading flow-cells with a higher than normal concentration of DNA.

Removal of other sequencing artifacts. We have also observed cases where the same end of the read-pair was sequenced twice (and consequently had a mapping distance of 0 or 1bp between the two ends). This artifact presumably arises from self-priming events during PCR. We excluded such matepairs from our analysis. Additionally, our DNA libraries often had a second minor peak around 100-200bp. This is an unintended artifact of library preparation. We therefore excluded all discordant mappings that were in F/R orientation and had a mapping distance <= 500bp. Consequently, we are unable to detect small insertions of novel sequence in this study. Lastly, we excluded all matepairs where both ends mapped within annotated SSR repeats. This minimizes false positives due to polymorphic SSRs, and reduces the number of mappings that must be examined.

Structural variation discovery with HYDRA

We developed HYDRA, a novel SV discovery algorithm, to identify SV in both unique and repetitive regions of mammalian genomes. Unlike most extant SV discovery algorithms (Chen et al. 2009; Korbel et al. 2009; Sindi et al. 2009), HYDRA compares multiple mappings from discordant matepairs to one another, and identifies putative SVs as those having a minimal number (two or more in this study) of matepairs with corroborating genomic positions, sizes and read orientations. A fundamental advantage of utilizing multiple mappings is that it permits the discovery SV in duplicated genomic regions such as segmental duplications (also known as low-copy repeats, or LCRs) as well as novel insertions of repetitive DNA (e.g. transposable elements, or TEs). Another less appreciated advantage of this approach is that the discovery of repeat insertions is not dependent on genome annotations.

HYDRA is written in C++ and uses data structures and algorithms from the Standard Template Library (STL). HYDRA identified SV among the 34.5 million discordant mappings (519,000 discordant matepairs) from the DBA individual in less than 5 minutes on a single processor while consuming less than 2GB RAM.

Preliminary screening for putative SV. HYDRA's speed comes largely by performing an efficient initial screen of all discordant mappings in search of evidence for potential SV. The four primary steps in this screening process are as follows:

a. We first determine which discordant mappings from each matepair should be retained for further SV discovery. Hydra allows one to retain: 1) the mappings with the *least edit distance* (termed "best" mappings), 2) all mappings within a user-defined edit distance of the "best" mappings, or 3) *all* mappings regardless of edit distance. In this study, we retained only the "best" mappings.
b. We then group all remaining discordant mappings where the ends of the matepairs are aligned to the same chromosome(s) and in the same orientation(s). This preliminary screen segregates similar discordant mappings that together would corroborate a potential SV, thereby greatly reducing the number of mappings that must be directly compared to one another in order to detect an SV "cluster".
c. We then sort each group of mappings from step (b) by the mapping distance between each end of the mapping (i.e., the mapping "length"). Once the mappings are sorted by length, we collect mappings whose lengths differ by no more than a user-specified "length deviation" (termed "*maxLengthDev*") parameter, which is based on the insert size variation of the sequencing library. Specifically, for any two mappings *i* and *j*, we require:

$$abs(length(i) - length(j)) \leq maxLengthDev$$

All mappings whose lengths meet this restriction are grouped into putative SV clusters. At the end of this step, Hydra has constructed clusters of mappings whose chromosome(s), orientation(s) and mapping lengths suggest potential SV. In the present study, the *maxLengthDev* parameter used was 2696bp, which represents 10 times the median absolute deviation (MAD) of the fragment lengths observed in our most variable DNA library.

d. The mappings within each putative cluster created in step (c) are then sorted by their genomic coordinates. This step further refines putative clusters by requiring that discordant mappings localize to the same genomic region(s) and thus support the same putative SV breakpoint. Once mappings in each cluster are sorted by their genomic coordinates, Hydra refines putative clusters by screening for mappings that span a common genomic interval and do not exceed a user-specified "non-overlap" (termed "*maxNonOverlap*") parameter, which is based on the insert size variation of the sequencing library. Specifically, for any two mappings *i* and *j* in a putative cluster, we require:

$$\left(abs(i.leftStart - j.leftStart) + abs(i.rightEnd - j.rightEnd)\right) \leq maxNonOverlap$$

where *leftStart* is the leftmost coordinate of each mapping and *rightEnd* is the rightmost coordinate of each mapping. As illustrated in Fig. S9, this restriction is designed to prevent the clustering of discordant mappings that have similar lengths yet do not support the same SV breakpoint. At the end of this step, Hydra has identified putative SV clusters from mappings that have similar lengths and orientations and support the same potential SV. In the present study, the *maxNonOverlap parameter* used was 2070bp. This is based on the fragment size variability observed in our most variable DNA library and represents 2 times its median fragment length plus 10 times its MAD.

While the current version does not yet account for multiple DNA sequencing libraries, this framework can easily be extended to multiple samples and libraries. Such an extension can be used to mix differing fragment sizes to increase breakpoint resolution. Moreover, it would enable multiple individuals and libraries to be combined for greater detection and genotyping sensitivity.

Refining SV breakpoints. After the preliminary screening for putative SV, clusters having a sufficient number of supporting matepairs (in this study a minimum of two matepairs) are further processed in an effort to choose the best set of mappings with which to describe the SV breakpoint. First, HYDRA compares each mapping (*i)* in each cluster to all the other mappings (*j*) in that cluster and tabulates how many other mappings meet both the *maxLengthDev* and *maxNonOverlap* restrictions with respect to the *i*th mapping. Mappings that meet both restrictions with respect to *i* are classified as "supporting" the *i*th mapping. The pseudocode below details this comparison:

```
for each mapping i in cluster:
     i_support = 0
     for each mapping j in cluster:
          if (i <> j):
               if (passesLengthDev(i,j) and passesNonOverlap(i,j)):
                    i_support = i_support + 1
               end_if
          end_if
     end_for
     update support for i to i_support
end_for
choose mapping with max(i_support) as seed
```

Hydra chooses the mapping that has the most “support” from the other mappings in the cluster as the “seed” mapping for the variant. Proper seed mapping selection maximizes the resolution of the putative breakpoint by incorporating the most supporting mappings. The variant is refined by iteratively adding the mapping with the next most support until we encounter a mapping that does not support all of the previously-added mappings. We are ultimately left with a set of discordant mappings that mutually corroborate the same SV and whose mappings collectively define the breakpoint of the variant.

Resolving ambiguities arising from multiple mappings. Since HYDRA may interrogate multiple mappings per discordant matepair, there are cases where one or more of the mappings for a given discordant pair support multiple structural variants. In such cases, we select the SV call that is supported by the most discordant mappings. In cases where multiple competing SV calls have the same level of mapping support, we select the variant with the least number of mismatches and gaps among all of the supporting mappings. In cases of a tie, a variant mapping location is selected randomly. Thus the final set of putative variants are those with the strongest support from the discordant mappings. Importantly, we also report those variants whose supporting mappings were redistributed to other more well-supported variants so that inter-sample variant comparisons can be made.

Excluding SVs that arose because of assembly errors in the reference genome. A primary concern for all current studies investigating SV via paired-end mapping approaches is false variant discovery owing to reference genome assembly errors. In this study, we had the advantage of sequencing a B6 mouse that is at most 30 generations separated from the mice sequenced to create the reference genome. Therefore, we were immediately suspicious of any putative SV that was observed in the B6 individual. Such events are most likely indicative of systematic alignment artifacts caused by misassembly.

Therefore, we excluded all putative SV in the DBA individual that were corroborated by at least one discordant mapping (i.e. identical orientation, and approximately the same mapping distance) in the B6 individual. The remaining set of variants were observed only in the DBA strain, and are therefore cleansed for assembly artifacts as much as possible given the data available.

Annotating HYDRA structural variant calls

Merging HYDRA breakpoint calls into non-redundant SV calls. To create a non-redundant set of SV calls, we combined HYDRA breakpoint calls that were within 4766bp (i.e., 2*(median fragment length + 10*MAD) for the most variable DNA fragment library) of one another and mutually supported the same class of mutation (e.g. inversion).

Intersecting SV with known genome features. We have developed a new software suite (BEDTools: http://code.google.com/p/bedtools) (Quinlan and Hall 2010) to facilitate the annotation and functional characterization of the SVs discovered in this study with respect to genome annotations in the UCSC Genome Browser's BED format (Kent et al. 2002). We created our own tools for such analyses because existing methods such as Galaxy (Giardine et al. 2005) and the UCSC Genome Browser were not amenable to our large datasets nor to complex queries involving several annotations. Notably, BEDTools includes a novel utility for screening for overlaps between SVs found via paired-end sequences and genome annotations in BED format.

Segmental duplication annotations. Segmental duplications (SD) in the mouse genome (She et al. 2008) were converted from mm8 (build 36) to mm9 (build 37) coordinates using the "liftover" utility provided by the UCSC Genome Browser. SV were declared to overlap with SD if at least 50% of the length of the variant overlapped with a SD or if either of the two ends of the variant overlapped with SD.

Definition of "recent" transposons. Annotated transposons for the mm9 build of the mouse genome were obtained from the UCSC Genome Browser's RepeatMasker (Smit et al. 1996-2004) table. Recent transposons were defined as those transposons that were less than 20% divergent (i.e. milliDiv <= 200) from the canonical transposon sequence for each class. Recent transposon insertions in the B6 lineage were defined as putative HYDRA deletions where at least 50% of the genomic span was comprised of a single recent transposon class (e.g. LINE1, or L1). Transposon insertions in the DBA lineage were defined as distant insertions where one end of the HYDRA variant overlapped with a recent transposon annotation. Cases where both ends overlapped with a recent annotation or one end overlapped with multiple recent annotations were classified as ambiguous.

Simple repeat annotations. We defined simple sequence repeats (SSRs) to be the union of the RepeatMasker track's "Simple_repeat" entries, the UCSC "Simple Repeats" track and the UCSC "Microsatellites" track.

Genes, exons, promoters. Gene and exon annotations were obtained from the UCSC RefSeq track. Promoter regions were defined from the RefSeq annotations by adding 1kb upstream of the transcription start site. Genes with known phenotypes were obtained from the UCSC "MGI Phenotypes" track. SV were declared to overlap with genes, exons, promoters or known phenotypes if the length of the variant shared at least one common base pair with a given annotation.

Calculation of SV enrichment in existing annotations. We classified SV as overlapping with genome features if they intersected by at least one base on either strand. Enrichment in segmental duplications was determined using a permutation experiment whereby segmental duplications were randomly shuffled among the genome (more specifically, the autosomes plus chromosome X) while their original sizes were maintained. We performed 1,000 such permutations and compared the mean number of SVs that overlapped with segmental duplications (by the method described above) in the permutation experiments to the observed number of overlaps with the true segmental duplication annotations. The P-values for enrichment reflect the fraction of permutations having more than the observed overlaps with segmental duplications.

Comparison to CNVs. We use the UCSC lift-over tool to convert published CNVs from the mm8 to the mm9 assembly. We classified a HYDRA variant as agreeing with a CNV reported by a previous aCGH study (She et al. 2008; Cahan et al. 2009) or depth of coverage (DOC) analysis (this study) if their genomic coordinates overlapped by more 10% with each other. This definition requires reciprocal overlap, such the shared interval comprises at least 10% of both the HYDRA variant and the CNV. We decided upon this lenient measure of overlap based upon the known imprecision of aCGH/DOC. CNV coordinates reported by aCGH depend heavily on the positions and local density of probes, and CNV coordinates reported by DOC depend upon the positions of non-overlapping 5kb windows. We noticed many cases where HYDRA variants and aCGH/DOC appeared to detect the same underlying variant, but showed relatively little overlap based upon the reported genomic coordinates. We performed this analysis using our final high-confidence dataset, as well as the complete dataset including low-confidence variants and alternative mappings for variants called at non-unique genomic regions. This latter comparison is crucial because often the genomic coordinates reported in multi-copy sequence can vary merely due to methodological differences in how variants are reported.

WGS "long-read" sequence alignment

Obtaining sequence data. We obtained 38,151,082 and 7,998,826 whole-genome shotgun (WGS) traces from the NCBI Trace Archive for the B6 and DBA strains, respectively. While there is no single accession number for these data, they can be retrieved using the following query: species_code='MUS MUSCULUS' and strain='C57BL/6J' and trace_type_code='WGS' and load_date <= '09/01/2009'. The DBA data can be retrieved using the same query substituting "DBA/2J" for "C57BL/6J".

Vector and quality trimming. We trimmed all traces prior to alignment according to the vector and quality trimming coordinates provided by the NCBI Trace Archive annotation files. Any sequence that was less than 100bp in length after quality and vector trimming was excluded from analysis. We were left with 34,624,688 and 7,998,824 reads for B6 and DBA, respectively.

Alignment with BLAT. We aligned all trimmed WGS long-reads for DBA and B6 with BLAT (Kent 2002)(version 32x1) using the following parameters: tileSize=12, stepSize=6, minMatch=4, minIdentity=90, minScore=30, extendThroughN, noTrimA, and maxIntron=100.

Identifying concordant and discordant reads. Long-reads from both B6 and DBA were classified as either mapping concordantly or discordantly with the reference genome. In order to be classified as concordant, we required that a given long-read have at least one mapping where 90% of the read aligned in a single block (with at most a 100bp gap, hence the maxIntron=100 BLAT parameter) and that 90% of the bases in the aligned portion matched the bases in the reference genome. All long-reads that failed this check were classified as discordant. We recorded all mappings for all discordant reads.

Validation of HYDRA variant calls

HYDRA identified 15,690 SV breakpoints between the DBA and reference genomes based from 34.5 million discordant Illumina matepair mappings. True structural differences between the DBA genome and the reference genome should be corroborated by the DBA long-reads but not the B6 long-reads. Specifically, bona fide SV in DBA should be supported by at least one DBA long-read that aligns as a so-called "split-read" alignment. For example, in the case of a true deletion in DBA (Fig. 2A,B), a DBA long-read should align such that two distinct portions of the read map to the regions of the reference genome that flank the deleted sequence in DBA. No such "split-read" alignment should be observed with the B6 long-reads.

We developed a pipeline to screen each putative DBA SV for supporting split-read alignments (split-reads) from both DBA and B6 long-reads. However, the HYDRA SV calls typically do not map to the exact breakpoint(s) of a given SV whereas the split-reads do. Therefore, we allowed an additional interval beyond the predicted HYDRA breakpoint to be examined for overlapping split-reads.

Specifically, for each putative SV, we computed the mean of the medians of each DNA library from which each supporting read-pair originated. This "mean of medians" was computed individually for each SV and was added to the interval that was examined in search of split-read alignments in B6 and DBA. We then required any observed split-read in B6 and DBA to have 90% overlap with the refined HYDRA SV intervals in order to be included in our SV validation scheme.

Using the above criteria, we classified variants where at least one DBA split-read was observed and no B6 split-reads were observed as confirmed. Variants where at least one B6 split-read was observed were classified as refuted. Owing to insufficient WGS coverage for DBA (but not B6), there were often cases where split-reads were not observed in either strain. Such cases were classified as inconclusive. These are very strict criteria for validation given that there was nearly 9-fold WGS coverage for the B6 strain.

The reader may note that there are other potential validation methods. One obvious alternative would be to allow concordant DBA long-reads that map to the predicted breakpoint interval to refute a putative SV. However, this approach suffers from one critical flaw: long-reads originating from duplicated or repetitive regions of the genome, which may in fact not be annotated as such due to their presence in the DBA genome but not the reference, can map to *bona fide* breakpoints in concordant fashion. We indeed tested a validation approach that utilized concordant long-read mappings in this way, and we noticed that true variants involving segmental duplications or transposons were often falsely refuted. We judged such variants as true based upon our ability to assemble and interpret their breakpoint sequences (see below). Nevertheless, even using a method that incorporates concordant read mappings HYDRA achieves a similar validation rate for simple SVs in unique genomic regions (>90%) (Fig. 2), and lower yet respectable rate for TEVs and multi-copy variants (60-80%, depending on the precise criteria).

Breakpoint assembly

Assembly of split-read WGS sequences that confirm HYDRA SV. The HYDRA SV calls that were confirmed by DBA long-reads were further characterized in an effort to identify the exact nucleotide at which the SV breakpoint(s) occurred. In such cases, we assembled the corroborating long-reads with PHRAP (Phil Green, unpublished, http://www.phrap.org/) using default parameters (Fig. 3). When a single WGS read confirmed the HYDRA SV, we attempted to identify the breakpoint from that single read. We excluded cases where PHRAP was unable to assemble supporting WGS reads into a single contig.

Alignment of the assembled breakpoint-containing contig (breaktig) to the reference locus. The PHRAP contigs were then aligned to the genomic locus that HYDRA identified (adding 5kb upstream and downstream to include sequence flanking the putative breakpoint(s)) with MEGABLAST. We employed

very sensitive settings (-s 0 -G 8 -E 2 -m 8 -W 8 -F f) to ensure that all homology between the assembled breaktig and the reference locus would be detected.

Selection of the "best" mappings. As a consequence of the sensitive alignment setting used, there are frequently cases where the same sections of a breaktig aligns to multiple locations within the reference genome locus. This occurs, for example, when a breaktig contains a common repeat. In such cases we retained the largest alignment for a given section of the breaktig. This served to eliminate secondary alignments that were not needed to characterize the SV breakpoint.

Calling and annotating SV breakpoints. The best alignments between each breaktig and the reference locus were used to classify SV breakpoints. Transposon insertions (TEVs) were identified as cases that appeared to be deletions (both flanking alignments were in the correct orientation, see Fig. 3B) in DBA with respect to the reference genome, yet the supposedly deleted region was at least 50% comprised of recent TE annotations in the reference genome (see below). These cases suggest a TE insertion in the B6 lineage rather than a deletion in the DBA lineage. We excluded all remaining breakpoints that identified a variant less than 100bp in size as well as cases that were clearly caused by an expansion of an annotated SSR. The remaining breakpoints were classified as deletions, duplications or inversions (Fig. 3B).

For all TEV and non-TEV breakpoints, we estimated the amount of homology at the breakpoint by computing the degree of "overlap" between adjacent alignments. When little or no overlap was observed, we classified the SV as a "flush" breakpoint. When significant negative overlap (i.e. there was a gap in the breaktig between two adjacent alignments to the reference genome, see Fig. 3B) was observed, we investigated the potential that DNA was inserted in the DBA genome at the breakpoint. Positive overlap indicates local sequence homology at the breakpoint. Substantial (e.g. >=20bp) sequence homology is indicative of NAHR, while so-called microhomology (e.g. <20bp) is indicative of either NHEJ, target-site duplication (TSD) caused by retrotransposon insertions, or replication-based template switching mechanisms such as FoSTeS or MMBIR.

Characterizing the origin of NAHR homology. We manually inspected those breakpoints that exhibited substantial homology to understand what types of sequence contribute to NAHR. In such cases, we required that both flanking alignments of a given breakpoint intersect with annotated segmental duplications, or recent LINEs, LTRs, SINEs or SSRs. Close inspection of cases where none of these sequence annotations were found revealed that there is in fact local homology, yet no annotated repeats.

Characterizing the origin of putative breakpoint insertions. We also inspected cases where there was evidence that additional sequence was inserted at the SV breakpoint in the DBA genome. For all predicted insertions larger than 20bp, we realigned the breaktig with both BLAT and BLAST (Altschul et al. 1990) and manually determined whether or not the inserted sequence was a true insertion, a complex rearrangement that appeared as an insertion, or whether it was merely an SSR expansion or alignment artifact. For all true insertions, we further determined whether the origin of inserted DNA was local (i.e. <10kb away), distant (i.e. >10Kb away or from a different chromosome) or foreign (i.e. no significant alignment was found in the mouse genome).

CNV discovery by depth of coverage analysis (DOC).

Removing GC-bias. For a given genomic interval the local depth of sequence coverage should be directly proportional to DNA copy number. Unfortunately, Illumina datasets suffer from GC-bias, such that local coverage depth is inversely related to local GC content. This source of noise can overwhelm the signal produced by CNV (Fig. S5). This noise can be effectively "cancelled out" by comparing similar datasets directly to one another, however GC bias can vary between datasets and a suitable control may not always be available. We therefore devised a method to compare a single dataset to the reference genome. Our approach is based upon the observation that, within a given GC content range (e.g. 40-40.5%), coverage is well approximated by a normal distribution (Fig. S5A). By dividing the genome up into "windows" of similar GC-content, depth of coverage can be assessed in a statistically straightforward manner. We first exclude reads that map to simple sequence repeats (SSRs), where abundant polymorphism can introduce local fluctuations in coverage. We then choose a genomic window size that contains an average of ~75-100 mappings (for this study 5kb) and fit a normal curve to the distribution of read counts in all windows within a given GC content range (0.5% intervals) using the MATLAB "normfit" function. To limit the effect of outliers on GC normalization, we exclude windows with read counts greater, or fewer, than the median read-count for that GC range plus or minus 4 median absolute deviations, respectively. We then calculate a Z-score, which is the number of standard deviations from the mean coverage of all of the 5kb windows with a given GC content. We use the Z-score for downstream analyses. We have compared the Z-score measurements to $\log_2$ ratios obtained by aCGH (Egan et al. 2007), and the DOC datasets are of similar if not superior quality (data not shown).

CNV identification. We identified CNVs using the same Hidden Markov Model (HMM) segmentation algorithm that we used previously for aCGH data (Egan et al. 2007). This model is designed to detect relative differences in DNA copy number between two genomes, in this case an Illumina-sequenced genome and the reference genome, when a copy number difference affects multiple adjacent measurements (i.e., oligonucleotide probes or windows of sequence coverage). Briefly, the model has

3 states: duplicated ("up"), equivalent ("ground"), and deleted ("down"). We assume that the Z-score of each 5kb window is generated from one of three Gaussian distributions N_u, N_g and N_d representing up, ground and down respectively. We take into consideration two sources of noise. A window could be part of a CNV while its Z-score belongs to the ground distribution. This can occur due to small-scale sequence variations that perturb matepair alignment, or to random noise. Similarly, the Z-score of a window in the ground state can belong to the distribution of a polymorphic state, perhaps due to mapping "pile-ups" at unannotated SSRs. Each state of the HMM therefore represents a Gaussian mixture of N_u, N_g and N_d with different mixture proportions for each state.

To obtain prior probabilities for the HMM we used a simple sliding window segmentation scheme. We first identify individual windows whose P-value for being in either the "up" or "down" state is beneath a given threshold, T_{seed}. We then explore adjacent windows by extending outward from the "seed", and continue to extend the segment so long as the P-value of two adjacent segments do not both exceed a second threshold (T_{extend}). T_{extend} is obtained by multiplying T_{scale} by the number of adjacent windows involved in a segment. By scaling T_{extend} to segment length it is possible to identify large CNVs with relatively subtle copy number differences (e.g., a 5/4 ratio). For our study we used T_{seed} =0.002 and T_{scale}=0.01.

We used the Vitterbi algorithm on the HMM to obtain the most probable state path. This path classifies genomic intervals as polymorphic (up or down) or not (ground). The segmentation was robust to small changes in the HMM parameters and the final set of CNVs were not very different to those obtained from the sliding window scheme. The HMM identified 178 segments in the B6 strain and 420 segments in the DBA strain.

CNV filtering. Since CNV discovery by DOC was not the main focus of our study, we sought to minimize false positives at the expense of false negatives. We therefore employed a strict CNV filtering scheme based upon the following confidence score:

$$score = (abs(Z_{cnv}) \times \ln(N)) - S$$

where:

Z_{cnv} = median Z-score of the identified CNV

N = number of consecutive windows that identify the CNV

S = standard deviation of the Z-scores among the windows of the identified segment.

We required that "up" CNVs had a confidence score greater than 4 and "down" CNVs greater than 3. We used these asymmetric thresholds due to the effects of poorly-annotated SSRs, which can cause local increases in sequence coverage. This filtering scheme removed 206 of the 598 (34%) CNVs identified by the HMM.

We also removed several classes of CNVs that appeared to be enriched for false positives. First, coverage depth at AT-rich regions of the genome can be inadequate for robust CNV discovery. We

thus removed 7 CNVs that had a mean GC content less than 35% and a mean read mapping count of less than 30 per 5kb window. We also discovered an interesting artifact related to the genomic distribution of repeats. We used the BWA mapping algorithm, which chooses a single mapping position randomly when multiple high-quality mappings are present. However, there exist a small number of loci that are so remarkably devoid of high-copy repeats that they are depleted in coverage and identified as "losses" relative to the reference genome. These loci have been previously described precisely for their exceptional lack of repeats (Prohaska et al. 2007). We thus manually inspected all CNV calls and removed the 10 that corresponded to such regions. These include the Hox gene clusters as well as a few other highly conserved loci. The HOX gene artifact is presumably the consequence of a reasonably large pool of retrotransposons copies that are not present in the reference genome assembly.

Finally, for the analyses discussed in the main text we did not include CNVs mapping to unplaced contigs ("random" chromosomes) since these are known to be misassembled, and since the sequences present in random chromosomes are generally also present at other genomic locations.

Genotyping. Since HMM segmentation is prone to false negatives we used more sensitive criteria to obtain "genotypes". For each CNV that was identified by the HMM and passed the above quality filters, but was only identified in one of the two strains, we examined the median Z-score of that interval for evidence of CNV in the other strain. If the median Z-score of a genomic interval exceeded 50% of the median Z-score of the CNV identified by the HMM, we scored it as a CNV. Of the 76 segments identified as different in both strains (relative to the reference genome) only 14 were identified by this genotyping criteria rather than the HMM. In addition, we only report "misassembled" loci at which the HMM called a variant in B6.

CNV validation by quantitative PCR

We randomly selected 5 HMM calls in the DBA strain that identified novel CNVs not reported by one of the two most comprehensive aCGH studies (She et al. 2008; Cahan et al. 2009). Primers for qPCR were generated using Primer3 (Rozen and Skaletsky 2000) under the conditions that the primer pair had a Tm of 58-60°C, runs of consecutive nucleotides were avoided, and the five nucleotides on the 3' end of the primer contained no more than 3 G/C bases. Primers were also analyzed for hetero- and homo-dimerization on the website of Integrative DNA Technologies. The amplicon size range was typically 50-150bp, and all primers were tested for specificity by amplification and gel electrophoresis preceding qPCR. Quantitative PCR was carried out in an Applied Biosystems 7300 Real Time PCR System. The 25µl reactions were composed of Applied Biosystems SYBR Green PCR Master Mix, 12.5ng of template DNA and 0.3µM of each primer. For each primer pair a reaction was set up for the query DNA, the reference DNA, and a control lacking DNA. All reactions were performed in 4 to 9 replicates. Each qPCR plate included a primer pair corresponding to a control locus known to be at

equivalent copy number in the query and reference DNA. We calculated the fold enrichment of the query DNA vs. the reference DNA at the locus of interest relative to the control locus with the following formula:

$$\frac{A\left(2^{(NTC-Ct_{query})} \Big/ 2^{(NTC-Ct_{ref})}\right)}{B\left(2^{(NTC-Ct_{query})} \Big/ 2^{(NTC-Ct_{ref})}\right)}$$

where:

A = Primer pair within the CNV.

B = Control primer pair in a region of equivalent copy number between reference and query DNA.

Ct = The threshold cycle value (a statistically significant increase in fluorescence).

NTC = No template control value.

This formula was applied separately to each reaction for a given primer pair in both the query and reference (ref) strains, in the order that the reactions were set up. The mean and standard error were calculated to generate Fig. S10.

Comparison of Hydra to VariationHunter-SC.

To assess the accuracy and sensitivity of Hydra's calls, we compared Hydra SV calls to those made by version 0.02 (October 19, 2009 release; downloaded from http://compbio.cs.sfu.ca/strvar.htm) of VariationHunter-SC (VH).

Creating an input file for VH. We developed a custom script to convert the discordant mappings used by Hydra to the input format required by VH. Since this version of VH does not call tandem duplications or inter-chromosomal events, we culled the entire file of 34.5 million discordant mappings to the 1,571,157 intra-chromosomal mappings that would suggest either deletions or inversions based on their mapping distance and orientations. This facilitated a direct comparison of deletion and inversion calls made by the two algorithms.

Parameters used for VH. To ensure that VH produced the most sensitive set of calls possible for this comparison, we required the "minimum weighted support for a cluster" parameter to be 1 and the "pre-processing mapping prune probability" parameter to be 0. These parameters allowed VH to call variants that are supported by 2 discordant mappings, yet have a "weighted" support less than 2 (personal communication, F. Hormozdiari). We then created a final set of VH calls consisting of all deletion (SV Type = 2) and inversion calls (SV Type = 3,4, or 5) that had mappings from 2 or more matepairs. In

total, VH called 6366 deletions and 525 inversions based on these criteria and ran in 4 minutes and 17 seconds.

Comparison to Hydra calls. We compared 6331 deletion and 495 inversion (all calls <= 1Mb in size) calls made by Hydra to the analogous calls made by VH. For this comparison, the Hydra deletions represent all intrachromosomal calls where the size and orientation suggest a deletion; therefore, these calls include events that we later annotated as transposon insertions in the reference genome. This was required for a direct comparison since VH did not further classify its deletion calls. When comparing Hydra and VH, we classified a variant as being called by both algorithms if there was at least 50% reciprocal overlap between the respective calls.

We compared Hydra's runtime on the same dataset (1,571,157 discordant mappings) as VH and found that Hydra ran ~13 times faster (19 second run time).

References

Altschul, S.F., Gish, W., Miller, W., Myers, E.W., and Lipman, D.J. 1990. Basic local alignment search tool. *J Mol Biol* **215**(3): 403-410.

Bentley, D.R. Balasubramanian, S. Swerdlow, H.P. Smith, G.P. Milton, J. Brown, C.G. Hall, K.P. Evers, D.J. Barnes, C.L. Bignell, H.R. et al. 2008. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature* **456**(7218): 53-59.

Cahan, P., Li, Y., Izumi, M., and Graubert, T.A. 2009. The impact of copy number variation on local gene expression in mouse hematopoietic stem and progenitor cells. *Nat Genet* **41**(4): 430-437.

Chen, K., Wallis, J.W., McLellan, M.D., Larson, D.E., Kalicki, J.M., Pohl, C.S., McGrath, S.D., Wendl, M.C., Zhang, Q., Locke, D.P. et al. 2009. BreakDancer: an algorithm for high-resolution mapping of genomic structural variation. *Nat Methods* **6**(9): 677-681.

Egan, C.M., Sridhar, S., Wigler, M., and Hall, I.M. 2007. Recurrent DNA copy number variation in the laboratory mouse. *Nat Genet* **39**(11): 1384-1389.

Giardine, B., Riemer, C., Hardison, R.C., Burhans, R., Elnitski, L., Shah, P., Zhang, Y., Blankenberg, D., Albert, I., Taylor, J. et al. 2005. Galaxy: a platform for interactive large-scale genome analysis. *Genome Res* **15**(10): 1451-1455.

Kent, W.J. 2002. BLAT--the BLAST-like alignment tool. *Genome Res* **12**(4): 656-664.

Kent, W.J., Sugnet, C.W., Furey, T.S., Roskin, K.M., Pringle, T.H., Zahler, A.M., and Haussler, D. 2002. The human genome browser at UCSC. *Genome Res* **12**(6): 996-1006.

Korbel, J.O., Abyzov, A., Mu, X.J., Carriero, N., Cayting, P., Zhang, Z., Snyder, M., and Gerstein, M.B. 2009. PEMer: a computational framework with simulation-based error models for inferring genomic structural variants from massive paired-end sequencing data. *Genome Biol* **10**(2): R23.

Li, H. and Durbin, R. 2009. Fast and Accurate Short Read Alignment with Burrows-Wheeler Transform. *Bioinformatics*.

Prohaska, S.J., Stadler, P.F., and Wagner, G.P. 2007. *Evolutionary Genomics of Hox Gene Clusters*. Springer, New York.

Quinlan, A.R. and Hall, I.M. 2010. BEDTools: A flexible suite of utilities for comparing genomic features. *Bioinformatics* **In Press**.

Rozen, S. and Skaletsky, H. 2000. Primer3 on the WWW for general users and for biologist programmers. *Methods Mol Biol* **132**: 365-386.

She, X., Cheng, Z., Zollner, S., Church, D.M., and Eichler, E.E. 2008. Mouse segmental duplication and copy number variation. *Nat Genet* **40**(7): 909-914.

Sindi, S., Helman, E., Bashir, A., and Raphael, B.J. 2009. A geometric approach for classification and comparison of structural variants. *Bioinformatics* **25**(12): i222-230.

Smit, A., Hubley, R., and Green, P. 1996-2004. RepeatMaster Open-3.0.

Waterston, R.H. Lindblad-Toh, K. Birney, E. Rogers, J. Abril, J.F. Agarwal, P. Agarwala, R. Ainscough, R. Alexandersson, M. An, P. et al. 2002. Initial sequencing and comparative analysis of the mouse genome. *Nature* **420**(6915): 520-562.

Zhang, Z., Schwartz, S., Wagner, L., and Miller, W. 2000. A greedy algorithm for aligning DNA sequences. *J Comput Biol* **7**(1-2): 203-214.