Simultaneous lineage tracing and cell-type identification using CRISPR-Cas9-induced genetic scars

Spanjaard, B., Hu, B., Mitic, N., Olivares-Chauvet, P., Janjuha, S., Ninov, N., Junker, J.P.

Nature Biotechnology
2018 MAI ; 36(5): 469-473
2018 APR 09 (first published online: final version)
doi: 10.1038/nbt.4124

URL: https://www.nature.com/articles/nbt.4124

Publisher: Nature America (Springer Nature)

Simultaneous lineage tracing and cell-type identification using CRISPR/Cas9-induced genetic scars

Bastiaan Spanjaard[1,3], Bo Hu[1,3], Nina Mitic[1], Pedro Olivares-Chauvet[1], Sharan Janjuha[2], Nikolay Ninov[2], Jan Philipp Junker[1,4]

[1] Berlin Institute for Medical Systems Biology, Max Delbrück Center for Molecular Medicine, Berlin, Germany

[2] DFG-Center for Regenerative Therapies Dresden, Technische Universität Dresden, Dresden, Germany

[3] equal contributions

[4] correspondence: janphilipp.junker@mdc-berlin.de (JPJ)

A key goal of developmental biology is to understand how a single cell transforms into a full-grown organism comprising many different cell types. Single-cell RNA-sequencing (scRNA-seq) is commonly used to identify cell types in a tissue or organ[1]. However, organizing the resulting taxonomy of cell types into lineage trees to understand developmental origin of cells remains challenging. Here we present LINNAEUS (LINeage tracing by Nuclease-Activated Editing of Ubiquitous Sequences)—a strategy for simultaneous lineage tracing and transcriptome profiling in thousands of single cells. By combining scRNA-seq with computational analysis of lineage barcodes, generated by genome editing of transgenic reporter genes, we reconstruct developmental lineage trees in zebrafish larvae, and in heart, liver, pancreas and telencephalon of adult fish. LINNAEUS provides a systematic approach for tracing the origin of novel cell types, or known cell types under different conditions.

Measuring lineage relationships between cell types is important for understanding fundamental mechanisms of cell differentiation in development and disease[2,3]. In early development and in adult systems with a constant turnover of cells, short-term lineage predictions can be computed directly on scRNA-seq data by ordering cells along pseudo-temporal trajectories according to transcriptome similarity[4-6]. However, the developmental origin of cells in the adult body cannot be identified using these approaches alone. Several approaches for lineage tracing exist. Genetically encoded fluorescent proteins are widely used as lineage markers[7,8], but due to limited spectral resolution, optical lineage tracing methods have mostly been restricted to relatively small numbers of cells. Pioneering studies based on viral barcoding[9,10], transposon integration sites[11], microsatellite repeats[12], somatic mutations[13,14], *Cre*-mediated recombination[15], and genome editing of reporter constructs[16,17] have used sequence information to increase the diversity of lineage labels. However, these methods have not been coupled with single-cell transcriptome sequencing and therefore do not provide any information on cell type.

Here we present LINNAEUS for simultaneous measurement of single-cell transcriptomes and lineage markers *in vivo.* The approach is based on the observation that, in the absence of a template for homologous repair, Cas9 produces short insertions or deletions at its target sites, which are variable in their length and position[16,18,19]. We reasoned that these insertions or deletions (hereafter referred to as genetic "scars") constitute heritable cellular barcodes that can be used for lineage analysis and read out by scRNA-seq (Fig. 1a). To ensure that genetic scarring does not interfere with normal development, we targeted an RFP transgene in the existing zebrafish line *zebrabow M,* which has 16-32 independent integrations of the transgenic construct[20]. Since these integrations are in different genomic loci (as opposed to being in tandem), we could make sure that scars cannot be removed or overwritten by Cas9-mediated excision. We injected Cas9 and an sgRNA for RFP into 1-cell stage embryos in order to mark individual cells with genetic scars at an early time point in development (Fig. 1b). Loss of RFP fluorescence in injected embryos served as a direct visual confirmation of efficient scar formation (Supplementary Fig. 1). At a later stage, we dissociated the animals into a single cell suspension and analyzed the scars by targeted sequencing of RFP transcripts (Online Methods). Simultaneously, we sequenced the transcriptome of the same cells by conventional scRNA-seq using droplet microfluidics[21] (Fig. 1c and Supplementary Fig. 2, 3).

We analyzed single cell transcriptomes of >70,000 single cells from dissociated larvae at 5 days post fertilization (dpf). On average, we detected ~3000 unique transcripts from ~700 detected genes per cell (Supplementary Data 1). Unsupervised clustering of single cell transcriptomes[22] revealed 70 groups of cells with distinct gene expression programs (Fig. 1d, Supplementary Fig. 4). We assigned these clusters to cell types based on differentially expressed genes (Supplementary Fig. 5, Supplementary Data 2, Online Methods). We found that Cas9 generated hundreds of unique scars per animal when targeting a single site in RFP (Fig. 1e, f, Supplementary Fig. 6), suggesting that analysis of genetic scars constitutes a powerful approach for whole-organism lineage analysis. Bulk analysis of 32 individual larvae revealed that some scar sequences are more likely to be created than others, probably through mechanisms like microhomology-mediated repair[23] (Fig. 1e). The scars with the highest intrinsic probabilities may be created multiple times per embryo and are therefore uninformative for lineage reconstruction. We therefore excluded the most frequent scars ($p>0.01$) from further analysis. We found that scarring continued until around 10 hours post fertilization, a stage at which zebrafish already have thousands of cells (Fig. 1g). Thus, our injection-based approach for Cas9 induction allowed us to label cells in an important developmental period during which the germ layers are formed and precursor cells for most organs are specified.

We detected variable numbers of scars in single cells, with the average number of scars per cell ranging from ~2 for erythrocytes to ~5 for epidermal cells. (Supplementary Data 3). This indicated that some lineage information was lost due to the sparsity of scRNA-seq data. To investigate this issue in more detail, we analyzed single cells from the offspring of Cas9-injected fish (Supplementary Fig. 7). In these fish, all cells (independent of the cell type) have the same scar profile, as they are derived from the same pair of germ cells. This analysis confirmed that scar detection efficiencies are cell type dependent, which probably reflects differences in cell size or promoter strength. Furthermore, we observed some variance in scar detection efficiency between the different transgenic integrations, which may be linked to genomic features of the integration sites. Notably, we did not find any highly expressed scars that were undetectable in specific cell types, suggesting that developmental silencing of specific integrations is no major concern.

To validate that genetic scars contain useful information about lineage relationships, we calculated enrichment or depletion of scar connections between pairs of cell types (Supplementary Fig. 8; Online Methods). Clustering cell types by scar connection strength revealed three groups, each of which contains either mostly ectodermal or mesendodermal cell types (Supplementary Fig. 9). We suggest this pattern was caused by a small number of scars that were created during the first cell divisions and then expanded locally. These groups of cell types form contiguous domains on the zebrafish fate map[24], but do not strictly correspond to germ layers, since the domain boundaries of scar clones do not necessarily align with the boundaries between germ layers.

Next, we set out to analyze the data at higher resolution and reconstruct lineage trees on the level of single cells instead of cell types. As our previous filtering of frequent scars removed scars that may have been created multiple times, the lineage tree should fulfil the maximum parsimony principle, with every scar being created exactly once. Indeed, maximum parsimony approaches have previously been used for inferring trees from CRISPR/Cas9 lineage data[16]. Earlier studies indicated that missing data does not need to be detrimental to maximum parsimony tree building methods[25]. However, such studies typically focused on a regime with an order of magnitude less taxa than we have cells, and more characters than we have scars. Using two simulated datasets, we found that Camin-Sokal maximum parsimony failed to reconstruct the correct tree for our system (Supplementary Fig. 10, 11). While it might be possible to solve this issue using modified versions of maximum parsimony or other established tree reconstruction algorithms, we developed an algorithm that is custom-tailored to our experimental system. Our custom-built strategy also facilitated integration of a filtering step to remove spurious connections. We therefore developed a computational method that fulfills the maximum parsimony criterion and allows for reconstruction of the correct tree in our system even if not all scars are detected in every cell (Supplementary Fig. 10, Online Methods, Supplementary Note 1). Our algorithm is based on the observation that there is a correspondence between the underlying lineage tree and the resulting scar network graph, a representation of all pairwise combinations of scars that are experimentally observed together in single cells (Fig. 2a). If all scar connections are detected, the scar that is created first has the most connections in the scar network graph, followed by scars that were created next, enabling lineage tree reconstruction in an iterative manner (Fig. 2b). To remove spurious connections, caused by cell doublets for example, scar connections that do not occur in enough cells were not taken into consideration (Online Methods, Supplementary Note 1). Using a simulated dataset with realistic parameters, we found that our computational method correctly reconstructed lineage trees (Supplementary Fig. 11). Finally, we placed all single cells in the lineage tree based on the detected scars (Fig. 2c). Scar dropouts meant that we did not have full lineage information about every single cell. However, the reconstructed lineage tree allowed us to infer a large part of the missing scar information (Supplementary Fig. 12). The resulting single cell lineage trees were then converted to a condensed representation for easier interpretation (Fig. 2d).

For the 5 dpf larvae we found that, as expected, the major developmental lineages shown in Fig. 1d separated at least partially from each other in the reconstructed lineage trees (Fig. 2e, Supplementary Fig. 13, 14). This data can be explored at different levels of granularity, and we decided to next focus on the cell types of the lateral plate mesoderm (Fig. 2f). We found that the different blood cell types have a shared lineage, but we observed that the erythrocytes are also found in an additional branch that does not contain any immune cells. This observation probably reflects the transition from primitive to definitive hematopoiesis in early zebrafish development, as primitive hematopoiesis produces mostly erythrocytes, whereas definitive hematopoietic stem cells are capable of generating all blood cell types[26]. The primitive and definitive hematopoietic stem cells are known to have different developmental origins. We found that the putative definitive hematopoietic cells have a shared lineage origin with endothelial cells (Fig. 2f, Supplementary Fig. 14), which is to be expected, as the definitive hematopoietic stem cells (but not the primitive ones) are derived from endothelial cells of the dorsal aorta. For endodermal and neuronal/neural crest cell types, we observed a similar structure of partially cell-type specific lineage branches (Supplementary Fig. 15). Due to the stochastic nature of cell labeling in LINNAEUS, scar creation is not synchronized with mitosis. It is therefore important to note that reconstructed lineage trees do not necessarily contain all cell divisions (Supplementary Fig. 11). Furthermore, early zebrafish development is highly variable[27]. We can therefore not expect to find exact correspondence of early lineage trees for all cell types in different animals.

In another set of experiments, we applied LINNAEUS to dissected organs of adult fish (Supplementary Data 4 and 5). Analysis of >40,000 cells from the telencephalon, heart, liver, and the primary pancreatic islet by scRNA-seq allowed us to identify many different cell types in these organs (Fig. 3a, Supplementary Fig. 16, 17). We first analyzed the resulting lineage trees at low granularity, which revealed a strong separation of the individual organs (Fig. 3b and Supplementary Fig. 18). However, we also detected several cell types, mostly from the immune system, that were present in multiple organs. We found that, as expected, the immune cells from different organs were grouped together in the lineage tree (Fig. 3c), which provided additional validation of our approach for scar filtering and lineage tree reconstruction.

We next zoomed into cardiac and pancreatic cell types (Supplementary Fig. 19). In agreement with literature, we detected an early separation of myocardial and endocardial lineages[28]. In the primary pancreatic islet, we observed scars that cover all three major endocrine cell types (alpha, beta, delta). However, we also found a smaller scar clone (scar 1204) in which delta cells are strongly underrepresented compared to the other scars, suggesting that the progenitors carrying this scar predominantly contributed to the alpha and beta cell lineages (Supplementary Fig. 19). Further studies would be necessary to corroborate potential biases of endocrine progenitors towards particular cell fates.

Related single cell lineage tracing methods based on CRISPR/Cas9 technology have recently been used to study brain development as well as the clonal history of different organ systems in the zebrafish[29,30]. An important advantage of CRISPR/Cas9 lineage tracing compared to competing technologies, such as viral barcoding and other inducible sequence-based lineage tracing methods, is the ability to move beyond clonal analysis and to computationally reconstruct full lineage trees on the single cell level. This is made possible by our computational approach for tree reconstruction that is robust to dropout events under realistic experimental conditions, and by our experimental strategy that uses independent scarring sites whose scars, once created, cannot be changed again. Within a single experiment, data analysis can be performed at different levels of granularity, from germ layers to organs and cell types. Our combined experimental and computational platform thus provides a powerful strategy for dissecting the lineage origin of uncharacterized cell types and for measuring the capacity of lineage trees to adapt to genetic or environmental perturbations. Our approach is based on an existing transgenic animal with multiple integrations of a transgenic construct, which should facilitate adaptation of the method to other model systems.

The observation that Camin-Sokal maximum parsimony failed to reconstruct the correct tree for our system (Supplementary Fig. 10, 11) serves as a cautionary note regarding computational analysis of CRISPR/Cas9 lineage data. However, additional studies would be necessary to systematically compare our algorithm to existing methods for tree reconstruction under different parameter regimes. Developing a general statistical framework for disentangling biological and technological variability of CRISPR/Cas9 lineage tracing remains another important open challenge for the future. We anticipate that future modifications of the experimental platform, such as for instance inducible systems, will enable longer periods of lineage tracing and molecular recording of cellular signaling events during cell fate decisions.

Methods

Methods and any associated references are available in the online version of the paper.

Acknowledgements

We thank R. Opitz, M. Guedes Simoes, D. Panakova, T. Durovic and J. Ninkovic for help with cell type identification. We also acknowledge support by MDC/BIMSB core facilities (zebrafish, genomics, bioinformatics), and we thank J. Richter for help with zebrafish experiments. Work in JPJ's laboratory was funded by a European Research Council Starting Grant (ERC-StG 715361 SPACEVAR), a Fondation Leducq Transatlantic Networks Grant (16CVD03), and a Helmholtz Incubator grant (Sparse2Big ZT-I-0007). BH was supported by a PhD fellowship from Studienstiftung des deutschen Volkes.

Author contributions

JPJ, BS and BH conceived and designed the project. BH, NM and SJ optimized dissection and dissociation protocols. BH developed the experimental approach for combined scar and transcriptome detection, and BH and NM performed experiments. BS developed computational methods and analyzed the data, with support by POC. JPJ and NN guided experiments, and JPJ guided analysis. JPJ and BS wrote the manuscript, with input from all other authors. All authors discussed and interpreted results.

Competing financial interests

The authors declare no competing financial interests.

References

1. Grün, D. & van Oudenaarden, A. Design and Analysis of Single-Cell Sequencing Experiments. *Cell* **163,** 799–810 (2015).
2. Woodworth, M. B., Girskis, K. M. & Walsh, C. A. Building a lineage from single cells: genetic techniques for cell lineage tracking. *Nature Reviews Genetics* **18,** 230–244 (2017).
3. Spanjaard, B. & Junker, J. P. Methods for lineage tracing on the organism-wide level. *Curr Opin Cell Biol* **49,** 16–21 (2017).
4. Trapnell, C. *et al.* The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nature Biotechnology* **32,** 381–386 (2014).
5. Setty, M. *et al.* Wishbone identifies bifurcating developmental trajectories from single-cell data. *Nature Biotechnology* **34,** 1–14 (2016).
6. Haghverdi, L., Büttner, M., Wolf, F. A., Buettner, F. & Theis, F. J. Diffusion pseudotime robustly reconstructs lineage branching. *Nat Methods* **13,** 845–848 (2016).
7. Barker, N. *et al.* Identification of stem cells in small intestine and colon by marker gene Lgr5. *Nature* **449,** 1003–1007 (2007).
8. Livet, J. *et al.* Transgenic strategies for combinatorial expression of fluorescent proteins in the nervous system. *Nature* **450,** 56–62 (2007).
9. Lu, R., Neff, N. F., Quake, S. R. & Weissman, I. L. Tracking single hematopoietic stem cells in vivo using high-throughput sequencing in conjunction with viral genetic barcoding. *Nature Biotechnology* **29,** 928–933 (2011).
10. Naik, S. H. *et al.* Diverse and heritable lineage imprinting of early haematopoietic progenitors. *Nature* **496,** 229–232 (2013).
11. Sun, J. *et al.* Clonal dynamics of native haematopoiesis. *Nature* **514,** 322–327 (2014).
12. Frumkin, D., Wasserstrom, A., Kaplan, S., Feige, U. & Shapiro, E. Genomic Variability within an Organism Exposes Its Cell Lineage Tree. *PLoS Comput. Biol.* **1,** e50–13 (2005).
13. Lodato, M. A. *et al.* Somatic mutation in single human neurons tracks developmental and transcriptional history. *Science* **350,** 94–98 (2015).
14. Ju, Y. S. *et al.* Somatic mutations reveal asymmetric cellular dynamics in the early human embryo. *Nature* **543,** 714–718 (2017).
15. Pei, W. *et al.* Polylox barcoding reveals haematopoietic stem cell fates realized in vivo. *Nature* **548,** 456–460 (2017).
16. McKenna, A. *et al.* Whole-organism lineage tracing by combinatorial and cumulative genome editing. *Science* **353,** aaf7907 (2016).
17. Frieda, K. L. *et al.* Synthetic recording and in situ readout of lineage information in single cells. *Nature* **541,** 107–111 (2017).
18. Junker, J. P. *et al.* Massively parallel clonal analysis using CRISPR/Cas9 induced genetic scars. *bioRxiv* (2017). doi:10.1101/056499
19. Schmidt, S. T., Zimmerman, S. M., Wang, J., Kim, S. K. & Quake, S. R. Quantitative Analysis of Synthetic Cell Lineage Tracing Using Nuclease Barcoding. *ACS Synth Biol* **6,** 936–942 (2017).
20. Pan, Y. A. *et al.* Zebrabow: multispectral cell labeling for cell tracing and lineage analysis in zebrafish. *Development* **140,** 2835–2846 (2013).
21. Klein, A. M. *et al.* Droplet Barcoding for Single-Cell Transcriptomics Applied to Embryonic Stem Cells. *Cell* **161,** 1187–1201 (2015).

22. Satija, R., Farrell, J. A., Gennert, D., Schier, A. F. & Regev, A. Spatial reconstruction of single-cell gene expression data. *Nature Biotechnology* **33,** 495–502 (2015).
23. Villarreal, D. D. *et al.* Microhomology Directs Diverse DNA Break Repair Pathways and Chromosomal Translocations. *PLoS Genet* **8,** e1003026–12 (2012).
24. Schier, A. F. & Talbot, W. S. Molecular genetics of axis formation in zebrafish. *Annu. Rev. Genet.* **39,** 561–613 (2005).
25. Wiens, J. J. Missing data and the design of phylogenetic analyses. *Journal of Biomedical Informatics* **39,** 34–42 (2006).
26. Jagannathan-Bogdan, M. & Zon, L. I. Hematopoiesis. *Development* **140,** 2463–2467 (2013).
27. Kimmel, C. B. & Warga, R. M. Indeterminate cell lineage of the zebrafish embryo. *Dev Biol* **124,** 269–280 (1987).
28. Keegan, B. R. Organization of cardiac chamber progenitors in the zebrafish blastula. *Development* **131,** 3081–3091 (2004).
29. **add reference** Alemany, A. et al. Whole-organism clone tracing using single-cell sequencing. *Nature.*
30. **add reference** Raj, B. et al. Simultaneous single-cell profiling of lineages and cell types in the vertebrate brain. *Nature Biotechnology.*

Figure legends

Figure 1. Using the CRISPR/Cas9 system for massively parallel single cell lineage tracing. (a) Cas9 creates insertions or deletions in an RFP transgene. These genetic scars can be used as lineage barcodes. Using the fish line *zebrabow M*, which has 16-32 integrations of the RFP transgene, enables us to record complex lineage trees with a single sgRNA. Simultaneous transcriptome profiling by scRNA-seq allows unbiased cell type identification. **(b)** Sketch of the experimental protocol. Injection of Cas9 and sgRNA for RFP into the zygote marks cells with genetic scars at an early developmental stage. Scars can be read out together with the transcriptome by scRNA-seq at a later stage. **(c)** Approach for simultaneous detection of scars and transcriptome from single cells. Cells are captured by droplet microfluidics, followed by lysis, reverse transcription, and amplification. After amplification, the material is split and processed into a whole transcriptome library and a targeted RFP library for scar detection. (**d**) t-SNE representation of scRNA-seq data and identified cell types for dissociated zebrafish larvae (5 dpf, n=7 animals). Cell types were grouped into 8 categories as indicated by the color code. **(e)** Probability distribution of scars, measured in bulk experiments on the DNA level. Pie chart shows fractions of different types of scars (deletion, insertion, single nucleotide polymorphism (SNP), complex scars). **(f)** Length distributions for deletions and insertions for the data shown in (e). **(g)** Scarring dynamics as measured on the DNA and RNA level, with exponential fit.

Figure 2. Computational reconstruction of lineage trees on the single cell level. (a) In a developmental lineage tree (top), each scar can be identified by a unique number corresponding to its ranking in the bulk scar frequency distribution (Fig. 1e). Newly created scars are indicated in black font. The resulting scar tree (middle), a reduced representation of the order of scarring events, can be represented as a network graph (bottom). In a scar network graph, each node corresponds to a different scar, and pairs of scars that are co-expressed in single cells are connected by gray lines. In LINNAEUS, we experimentally measure scar network graphs, based on which we computationally reconstruct the underlying lineage tree. **(b)** Cartoon of the computational approach. Network graphs allow reconstructing the order of scar creation events in an iterative approach. The first scar is determined as the one with the highest connectivity (red arrow). Upon removal of the first scar and its connections,

the following scars are identified as the most highly connected ones in the reduced network. For details see Online Methods. **(c)** After the scar tree has been built, we position all individual cells in the tree according to their scar profile. Incomplete scar detection efficiency may lead to loss of information in single cells (black numbers: detected scars; gray crossed out numbers: missed scars). As a consequence, some cells cannot be placed all the way down to the lowest branch of the tree (example: red cell, in which scar 41 and 75 were not detected). However, some missing scars can be reconstructed (example: blue cell, in which scar 41 can be inferred). See also Supplementary Fig. 12. **(d)** Sketch of a simple single cell lineage tree with two cell types (red, blue). Single cell lineage trees can be represented in a condensed form by indicating fractions of cell types as pie charts (cumulative with respect to the branches below). **(e)** Lineage tree for one 5 dpf larva. Pie charts are plotted small for n<50, medium for n≥50, and large for n≥1000. Color code for cell types as in Fig. 1d. Scars with creation probability ≥0.001 and scars that were detected in more than 1 larva were excluded from the analysis. In general, developmental lineages separate well in the tree. However, since scarring ends at ~10 hours post fertilization, the end points of the branches may still give rise to multiple cell types in multiple tissues. **(f)** Lineage tree for one 5 dpf larva, zoomed into lateral plate mesoderm (see color code). The tree structure was determined based on the whole dataset (e).

Figure 3. Single cell lineage analysis of adult organs reveals hierarchies of cell fate decisions. (a) t-SNE representations of scRNA-seq data for dissociated organs from adult zebrafish (red: heart, green: pancreas + liver, blue: telencephalon; n=3 animals). **(b)** Lineage tree for organs from one adult. Pie charts are plotted small for n<50, medium for n≥50, and large for n≥1000. Scars with creation probability ≥0.01 were excluded from the analysis. Color code as in (a). **(c)** Lineage tree zoomed into immune cell types from same adult as (b) (see color code). As expected, immune cells from different organs cluster together in the lineage tree, even though the sequencing libraries for the different organs were prepared separately. This observation is an additional important validation of the scar filtering pipeline, since it shows that even small cell populations such as these immune cells do not acquire scars from other cells types in their organ of origin by mechanisms such as cell doublets or sequencing errors.

Online Methods

Zebrafish lines and animal husbandry

We used the transgenic zebrafish line *zebrabow M*[20] for LINNAEUS. This line has multiple integrations of a transgenic construct that expresses RFP from the ubi promoter, which is constitutively active in all cell types. Fish were maintained according to standard laboratory conditions. All animal procedures were conducted as approved by the local authorities (LAGeSo, Berlin, Germany) under license number G0211/16. We set up crosses between *zebrabow M* adults with high RFP fluorescence, and we injected the embryos at the 1-cell stage with 2 nl Cas9 protein (NEB, final concentration 350 ng/µl) in combination with an sgRNA targeting RFP (final concentration 50 ng/µl, sequence: GGTGTCCACGTAGTAGTAGCGTTTTAGAGCTAGAAATAGCAAGTTAAAATAAGG CTAGTCCGTTATCAACTTGAAAAAGTGGCACCGAGTCGGTGCTTTT). Since injection efficiencies may vary (Supplementary Fig. 1), we selected embryos with low RFP fluorescence for single cell analysis. For control experiments in Supplementary Fig. 2 and 7 we set up crosses between pairs of adult Cas9 injected fish.

The sgRNA was in vitro transcribed from a template using the MEGAscript® T7 Transcription Kit (Thermo Scientific). The sgRNA template was synthesized with T4 DNA polymerase (New England Biolabs) by partially annealing two single stranded DNA oligonucleotides containing the T7 promotor and the RFP binding sequence, and the tracrRNA sequence, respectively. In the experiments described here, we did not use the ability of the line *zebrabow M* to switch from RFP to YFP or CFP expression upon addition of Cre[20].

Preparation of single cell suspensions

Single larvae at 5 dpf were transferred into 50 µl HBSS containing 1x TrypLE™ (Thermo Fisher Scientific) and incubated at 33°C for ~20 minutes with intermittent pipette mixing (every 5 minutes) until the larva was no longer visible. 500 µl cold HBSS (Thermo Fisher Scientific) supplemented with 1% BSA was then added to the suspension, and the cells were pelleted in a table-top centrifuge at 4°C and 300 g for 5 minutes. The pellet was washed with 500 µl cold HBSS supplemented with 0.05%

BSA and centrifuged down again. The resulting pellet was resuspended in the same buffer and filtered through a cell strainer of 35 µm diameter.

Adult zebrafish were euthanized by an overdose of tricaine in combination with low water temperature. Afterwards, heart, brain, pancreas islets, and liver were isolated from the fish. Single cell suspensions of the organs were obtained using different protocols:

Heart: The zebrafish heart including atrium, ventricle and *bulbus arteriosus* was transferred into cold HBSS and opened carefully with forceps, allowing most of the erythrocytes to be washed away. Afterwards, the heart tissue was transferred into 500 µl HBSS containing Liberase™ enzyme mix (Sigma-Aldrich, 0.26 U/mL final concentration) and Pluronic® F-68 (Thermo Fisher Scientific, 0.1 %). The reaction was incubated at 37°C for 30 minutes while shaking at 750 rpm with intermittent pipette mixing. Afterwards, most of the tissue was dissociated. The reaction was stopped by adding 500 µl cold HBSS supplemented with 1% BSA. The cells were pelleted by centrifuging at 200 g in a table-top centrifuge at 4°C, then washed and filtered following the procedure described above for 5 dpf larvae.

Brain: The telencephalon without olfactory bulbs was isolated in cold HBSS and immediately transferred to a solution of HBSS with 0.81% D-glucose and 15 mM HEPES. Dissociation was initiated by adding 0.1x TrypLE™ and 0.1 % Pluronic® F-68 (final concentrations). The tissue was incubated for 30 minutes at 37°C while shaking at 750 rpm, with occasional gentle mixing. The dissociation reaction was stopped by addition of equal volume of EBSS solution containing 4% BSA and 20 mM HEPES. The sample was filtered using a 70 µm filter and centrifuged at 300 g for 5 minutes, after which the pellet was washed once with PBS and resuspended in HBSS with 0.04% BSA. Finally, the suspension was filtered with a 35 µm filter.

Pancreas and liver: The pancreatic tissue containing preferentially the primary pancreatic islet was isolated under a stereomicroscope and transferred into 500 µl HBSS containing 1x TrypLE™ and 0.1% Pluronic® F-68. The liver was isolated and dissected into small pieces, one of which was transferred into 500 µl HBSS containing 1x TrypLE™ and 0.1% Pluronic® F-68. After 30 minutes of incubation at 37°C with intermittent pipetting, the suspensions were pelleted, washed and filtered following the procedure described above for 5 dpf larvae.

All final single cell suspensions were quantified and controlled for quality by microscopy using a hemocytometer.

Scar detection in bulk samples

DNA-based scar detection: DNA of single animals was extracted by heating the samples in 50 µl of 50 mM NaOH at 95°C for 20 minutes. 1/10 volume of 1 M Tris-HCl, pH = 8.4 was then used to neutralize the mixture. We took 20 µl of the DNA for amplification of scar sequences using RFP-specific barcoded primers. The RFP primers were chosen such that the cut site of Cas9 was positioned approximately in the middle of the sequencing read. We then pooled the PCR products, performed a clean-up reaction using magnetic beads (AMPure Beads, Beckman Coulter), and added Illumina sequencing adapters in a second PCR reaction. Primer sequences are provided in Supplementary Table 1.

RNA-based scar detection: RNA of single or pooled animals was extracted with TRIzol™ Reagent (Thermo Fisher Scientific) according to the manufacturer's protocol. The RNA was precipitated using isopropanol, and the pellet was washed 2 times with 75% ethanol, air dried, and resuspended in 10 µl of reverse transcription mix (0.3 µM poly-T primer, 1x first strand buffer (Thermo Fisher Scientifc), 10 µM DTT, 1 mM dNTPs, 0.5 µl RNAseOUT™ (Thermo Fisher, Cat. No. 10777019), 0.5 µl SuperScript™ II (Thermo Fisher, Cat. No. 18064-014. The reaction was incubated at 42°C for 2 h for reverse transcription, followed by scar specific PCR amplification as described above for DNA-based scar detection.

Transcriptome and scar detection in single cells

Single cells were captured using Chromium™ (10X Genomics, PN-120233), a droplet-based scRNA-seq device according to the manufacturer's recommendations. Briefly, the instrument encapsulates single cells with barcoded beads, followed by cell lysis and reverse transcription in droplets. Reverse transcription was performed with polyT primers containing cell-specific barcodes, Unique Molecular Identifiers[31] (UMI), and adapter sequences. After pooling and a first round of amplification, the library was split in half. The first half was fragmented and processed into a conventional scRNA-seq library using the manufacturer's protocols. We used the second, unfragmented, half to amplify scar reads by two rounds of PCR, using two nested forward primers that are specific to RFP, and reverse primers binding to the adapter site. The RFP primers were chosen such that the cut site of Cas9 was positioned approximately in the middle of the sequencing read, ensuring that a broad range of deletion lengths can be reliably detected. Primer sequences are provided in Supplementary Table 1. We confirmed successful library preparation by Bioanalyzer (DNA HS kit, Agilent). Samples were sequenced on Illumina NextSeq 500 2x 75 bp and Illumina HiSeq 2500 2x 100 bp.

Mapping and extraction of single cell mRNA transcript counts

A zebrafish transcriptome was created with Cell Ranger 2.0.2 from GRCz10, release 90. Alignment and transcript counting of libraries was done using Cell Ranger. Cell numbers to be extracted were set at a minimum of 6000 but were increased if there were substantially more cells with more than 500 unique transcripts. Exact numbers can be found in Supplementary Data 1.

Mapping and filtering of single cell scar data

Scar reads have the same structure as transcript reads: they consist of a barcode, a UMI and a scar. The scar sequences were aligned using *bwa mem*[32] to a reference of RFP. Valid cell barcodes were identified based on the single-cell transcriptome data (see previous paragraph). We removed reads that were unmapped, had an incorrect barcode, or did not start with the exact PCR primer we used. We truncated all scar sequences to 75 nucleotides and filtered out shorter sequences.

To mitigate the effect of sequencing errors, we implemented several rounds of scar filtering (Supplementary Fig. 2). We started by counting the number of times each molecule was sequenced. Sequencing errors will typically have fewer reads than the actual scars they originate from. As a first filtering step, we therefore removed all molecules only seen once to reduce the complexity in the dataset for consecutive filtering steps.

In the second filtering step, we aimed to remove easily recognizable sequencing errors and chimeric reads[33]. To this end, we consecutively considered scar sequences that have the same cellular barcode and UMI, UMIs that have the same cellular barcode and scar sequence, and cellular barcodes that have the same UMI and scar sequence. In each step, we kept only the molecule with the highest number of reads. The rationale behind this is that it is very improbable to have two valid scar sequences in the same cell with the same UMI, or to have a scar sequence with the same UMI appear in two different cells. The observation of two different UMIs for the same scar in the same cell is much more likely and corresponds to detection of multiple transcripts from the same locus, but information about scar expression levels was not required in our downstream analysis.

In the third filtering step, we specifically targeted sequencing errors within each cell. We compared the scar sequences found within a cell to each other. We filtered out sequences that had a Hamming distance of 2 or less to another scar sequence in the same cell that occurred in at least eight times as many reads. Scar sequences in the same cell that were one Hamming distance apart but had a read ratio less than eight were tested on three criteria if both of them occurred at least twice in the scar library:

1. Do both scars have more than one transcript?
2. Do both scars occur in cells independently from each other?
3. Do the UMIs of both scars have Hamming distance of two or more?

If two of these criteria were true, the scars were kept and the sequences were placed on a list of validated scars that, if they occurred in the same cell in another library, did not have to be tested anymore. If one or zero criteria were true, the scar that had only one transcript, or the scar that did not occur independently, were filtered out.

In the fourth filtering step, we determined the distribution of reads for the scars we had kept so far. Based on this distribution we set a cut-off and filtered out the scars that did not have at least this number of reads. Finally, for each cell type we determined the distribution of different scars seen per cell and set a maximum number of scars a cell of that type can have. We filtered out cells in which we observed more than this maximum number as possible doublets.

While each scar is identified by its sequence, scars are labeled in the manuscript using their ranking in the bulk scar frequency distribution (e.g. “scar 77”) or their CIGAR code (e.g. “47M6D28M”) as a shorthand notation. Since scars cannot be modified once created, each scar is considered as a separate entity for lineage tracing independent of its sequence.

Determination of scar probabilities

We aligned reads from thirty-two single embryos (DNA-based bulk scar detection) to a reference of RFP. We filtered out unmapped reads and reads that did not start with the exact PCR primer, and truncated all reads to one hundred nucleotides, removing shorter ones. To determine the creation probabilities of the different scars, we removed all unscarred RFP reads from each embryo. We normalized the scar content of each embryo to one and calculated scar probabilities as the average ratio with which each scar was observed.

To account for the slightly different sequencing read structure of single cell and bulk scar detection (see above), we considered only the nucleotides that are shared between the two approaches, and we assigned the bulk scar probabilities to single cell scars accordingly. Single cell scars that were not detected in bulk had their probability set to the lowest probability value detected in bulk.

Determination of scarring dynamics

Embryos were injected with Cas9 and sgRNA at the 1-cell stage. After 1, 2, 3, 4, 6, 8, 10, and 24 hours, several embryos were collected and pooled (5-6 for earlier stages, 2-3 for later stages), followed by RNA and/or DNA extraction using TRIzol Reagent. Bulk scar libraries were produced as described above. For each sample, we calculated the percentage of unscarred RFP. We fit a negative exponential to this data, assuming that the fraction of unscarred RFP at t=0 was one.

Identifying cell types

We used the R package 'Seurat', version 2.1.0[22], for cell-type identification as described below. We removed genes that were not found in at least three cells, and removed cells that had less than two hundred of those genes. We log-normalized the transcript counts and removed cells with more than 2,500 genes observed. For single cells from 5 dpf larvae and adult pancreas, we filtered out cells with a mitochondrial content of more than 7.5 percent, and for single cells from adult hearts and telencephalons we filtered out cells with a mitochondrial content of more than fifteen percent; we expect the cardiomyocytes in particular to have high mitochondrial content.

We regressed out influences of the number of transcripts, mitochondrial transcripts, and libraries, and kept a total of 2779 highly-variable genes for cells of 5 dpf larvae, 3775 highly-variable genes for cells of adult telencephalon, 4536 for cells of adult heart and 3018 for cells of adult pancreas. We performed a principal component analysis and kept the first sixty components for single cells from 5 dpf larvae, eleven for adult brains, eight for adult hearts, and fifty for adult pancreases. Clustering, using the smart local moving algorithm[34] on a K-nearest neighbor graph of cells, was done on these components with resolution 1.8 for 5 dpf larvae, resolution 0.8 for adult brain cells, resolution 1.0 for cells from adult heart and adult pancreas. Dimensionality reduction, using t-Stochastic Neighbor Embedding[35,36] (tSNE), was done on the sixty components for the 5 dpf larvae, and on components three to twenty-two for the adult organs to reduce the visual impact of batch effects. To calculate differential gene expressions, we used the likelihood-ratio test as implemented in Seurat, introduced in McDavid et al., 2013[37], with an underlying negative binomial distribution for gene expression. This test aims to detect changes in mean gene expression and expression frequencies over different clusters. Using these differentially expressed genes, we assigned clusters to cell types based on literature and the ZFIN database[38] (Supplementary Data 2 and 4). We did not aim to identify all cell types with maximal resolution and focused instead on unequivocal identification of those cell types that are highlighted in the text (such as the larval hematopoietic cells, and adult pancreatic cells). Cell type assignments of all other clusters should therefore be considered tentative. Clusters were subsequently merged if they were found to have the same cell type, and we applied a mild coarse-graining by merging highly related cell types (for instance, different neuronal subtypes in the adult telencephalon were merged).

Connection enrichment analysis

We used an analysis of the scars shared between cells to illuminate the overall structure of the sequencing results from 5 dpf larvae. We expect that cells in which we observe the same scar have a shared lineage. To understand the scarring process better, we aimed to find out which cell types share many scars – these cell types would have a strong lineage relationship – and which cell types do not share many scars – these cell types would not have many immediate shared precursors.

We call cells 'connected' if they share at least one scar that has a creation probability of less than 0.1% and is only present in one organism. To find out whether cell types have a higher number of connections between them than expected by chance, we developed the background model described below (see also Supplementary Fig. 8). The background model starts with the realization that a connection is defined by its endpoints, and that therefore the number of expected connections between two cell types is determined by the number of connection endpoints of the two cell types. More precisely, the chance of forming a connection between cell type A and B is given by $p(A\text{-}B) = 2 * CE(A)*CE(B)/CE(tot)^2$, and that of forming a connection within cell type A by $p(A\text{-}A) = CE(A)^2/CE(tot)^2$, with $CE(A)$ the number of connection endpoints of cell type A, and $CE(tot)$ the total number of connection endpoints. These probabilities define a binomial background model. Using this model, we calculate the enrichment z-score between cell types, i.e. how many standard deviations the observed number of connections between two cell types is away from the expected number of connections. A positive enrichment score indicates more connections than expected by chance, a negative enrichment score indicates less connections than expected by chance.

We define the distance between cell types based on their enrichment z-scores by the following equation: $D(A, B) = 1 - (E(A, B) - E_{min})/(E_{max} – E_{min})$, with $D(A, B)$ the distance between cell types A and B, $E(A, B)$ the enrichment z-score between them, E_{min} the minimal enrichment z-score and E_{max} the maximum enrichment z-score. The term $E – E_{min}$ can be understood as a translation of all enrichment scores to positive values. These values are then divided by the maximum value and subtracted from 1 to create distances scaled between 0 and 1. We performed hierarchical clustering on these distances, using average linkage as implemented by the *hclust* function in R. We performed this analysis for two larvae, cutting the dendrogram into three and four clusters, respectively (Supplementary Fig. 9).

Tree building

Our computational method for lineage tree reconstruction consists of two phases. First, we derive the order of scarring events. To do so, we make use of scar network graphs, a representation of all pairwise combinations of scars that are experimentally observed together in single cells (Fig. 2a). If all scar connections are detected, the scar that is created first has the highest degree of connections in the scar network graph, followed by scars that were created next, enabling lineage tree reconstruction in an iterative manner (Fig. 2b). In the second phase, we place all cells in the lineage tree according to their scar profile (Fig. 2c). Cells are placed as low in the tree as their scars allow. Due to incomplete scar detection, we do not have full lineage information about every single cell. However, the structure of the scar network graph is robust towards scar dropouts, since it is based on the collective information of thousands of single cells (see simulations in Supplementary Fig. 10). To ensure that lineage tree reconstruction is not affected by known experimental biases, we also included the following measures:

- *Double scarring:* Some scars have a higher intrinsic probability than others (Fig. 1e). To minimize the chance of considering scars that may have been created twice or more in the same fish, we excluded all scars that have a probability higher than 0.1%. With this threshold, most scars were unique to a single fish among the replicates studied (Supplementary Fig. 6). Any remaining scars that were not unique to a single replicate were also excluded from the subsequent analysis.
- *Cell doublets:* Co-encapsulation of two cells in one droplet is a known limitation of scRNA-seq techniques that are based on droplet microfluidics. Cell doublets can lead to spurious connections between scars in the network graph. Incomplete tissue dissociation, limited barcode diversity, barcode sequencing errors, and free-floating RNA from cells burst in the microfluidic system may potentially have similar consequences. As a protection against this effect, we only accept connections in the scar network graph that are more highly detected than expected by chance given a library-specific doublet rate (typically around 10%, depending on the experimental cell loading rate). See Supplementary Note 1 for details.
- *Missing connections:* In case of very low cell numbers or scar detection efficiencies, it is possible that a connection is missed in the scar network graph. To address this issue, we performed a statistical test for each scar to check whether the number of observed connections is compatible with the scar being on top of the current sub-branch, given the numbers of cells and the observed scar dropout rates (Supplementary Fig. 20, Supplementary Note 1). In each

iteration, we tested only those scars whose inferred detection rate (if placed on top of the corresponding sub-branch) was higher than 0.1, a threshold derived experimentally in Supplementary Fig. 7.

- *Pruning the tree:* Especially for later, smaller branches, it is possible that not enough connections are observed to accurately place them in the lineage tree, resulting in positioning of the branch too high up in the tree (Supplementary Fig. 20, Supplementary Note 1). We prune the lineage tree for such branches by removing branches that have less than 25% of the cells their siblings have.

Using simulated data under realistic conditions (including cell doublets, as well as cell type and integration site dependent scar detection efficiencies), we demonstrate that this approach reconstructs the correct lineage tree (Supplementary Fig. 11). Lineage trees were visualized by expanding the R-package *'collapsibleTree'* (https://github.com/AdeelK93/collapsibleTree/) with previous authorization from the author. This package relies on the D3 javascript libraries.

Simulations

We simulated the scarring process during embryo development (Supplementary Fig. 10, 11). To do this, we used a simple model that starts with one cell, and in which all cells present undergo synchronized mitosis. Every cell cycle, the RFP integrations of the cells can acquire a unique scar. The chance of creating a scar is fixed for every integration for every cell division, and all scars are transmitted to the progeny of the cells.

After simulating the scarring events during development, we also simulate a sequencing experiment that produces data for tree building. To this end, the cells at the bottom of the tree are clonally expanded, generating many copies that all have the same scar profile. The experimental data consists of a sample of these cells, with a scar detection rate determining the chance of seeing a scar that is present in a cell.

We simulated two distinct trees. The first is a simple tree of three generations, where all cell divisions are marked by acquisition of new scars (Supplementary Fig. 10). From this tree we sampled 125 cells with a scar detection rate of 0.3, yielding 99 cells in which at least one scar was detected. This dataset was then used to compare LINNAEUS tree building with maximum parsimony tree building.

The second simulated tree was a more realistic tree in which six generations of cells can potentially receive a scar on ten target sites (Supplementary Fig. 11). Here, we used a cell division rate of 4 per hour, as measured by microscopy[39,40]. A scarring rate of 0.4 per hour reproduced the fit scarring dynamics during the first three hours (Supplementary Fig. 11a). We can use this simulation to estimate the number of new scars per cell division (Supplementary Fig. 11b). In this simulation, we assumed three cell types (fraction 15%, 25%, 60%) with different detection rates (70%, 30%, 10%, respectively). We furthermore assumed that two of the ten target sites are much harder to detect (by a factor 20, i.e. detection rates 3.5%, 1.5%, 0.5%). The resulting developmental tree is shown in Supplementary Fig. 11c. Due to the stochasticity of scar creation, scars are not created in all precursor cells, and in Supplementary Fig. 11d we show the maximal lineage tree that can be measured by scars. We expand all final branches (not shown) and sample 2000 cells from the resulting pool with a cell doublet rate of 5%, yielding 1716 cells (including doublet cells) with at least one scar.

Tree building on simulated data

To validate our tree building method, we built trees from both simulated trees using the cells sampled as described in the section “Simulations”. We compared our results to maximum parsimony tree building as done by the program “mix” in PHYLIP 3.695[41], using the Camin-Sokal algorithm with missing states encoded as “0”. If multiple trees were tied for best tree, we took the first generated tree.

The simple developmental tree (Supplementary Fig. 10) was recreated flawlessly by the LINNAEUS tree building algorithm (Supplementary Fig. 10b). However, maximum parsimony was not able to resolve the tree correctly, creating unjustified complexity due to multiple creation events for the same scar (Supplementary Fig. 10c). The more realistic scar tree (Supplementary Fig. 11) was also recreated faithfully by LINNAEUS (Supplementary Fig. 11f). Maximum parsimony again created a strong amount of unjustified complexity with a total of 265 scarring events for 46 scars, an average of over five times per scar (Supplementary Fig. 11g).

Statistics

We assessed RNA scar expression rates by comparing scar abundance in DNA to scar abundance in RNA in three 24 hpf animals. Using 70,251 data points, every one of which representing the RNA and DNA abundances of a sequence in one fish, we found a Pearson correlation of 0.97 between RNA and DNA abundances (Supplementary Fig. 3).

We used Seurat to identify cell types in four datasets: 72,252 cells from 5 dpf larvae (n=7 animals) and cells from three different organs in adult fish (n=3 animals): heart (12,248 cells), telencephalon 7,045 cells) and pancreas/liver (20,777 cells). Distribution of cell numbers over identified clusters can be found in Supplementary Data 2 (larvae) and 4 (adults). We determined differential gene expression using Seurat's "negbinom" test that includes a Benjamini-Hochberg correction of p-values.

To determine whether cell types had a statistically significant amount of connections (Supplementary Fig. 8 and 9), we first determined the theoretical connection probability of two cell types following the reasoning laid out above. We then used a two-tailed binomial test to assess whether the actual observed number of connections between the two cell types is different from the expected number of connections. The p-values were corrected for multiple testing using the Benjamini-Hochberg correction. Values for all 2,485 tests can be found in Supplementary Data 6.

Data availability

Sequencing data are deposited on Gene Expression Omnibus, accession number GSE106121. Interactive single cell lineage trees are available at http://bimsbstatic.mdc-berlin.de/junker/linnaeus/index.html

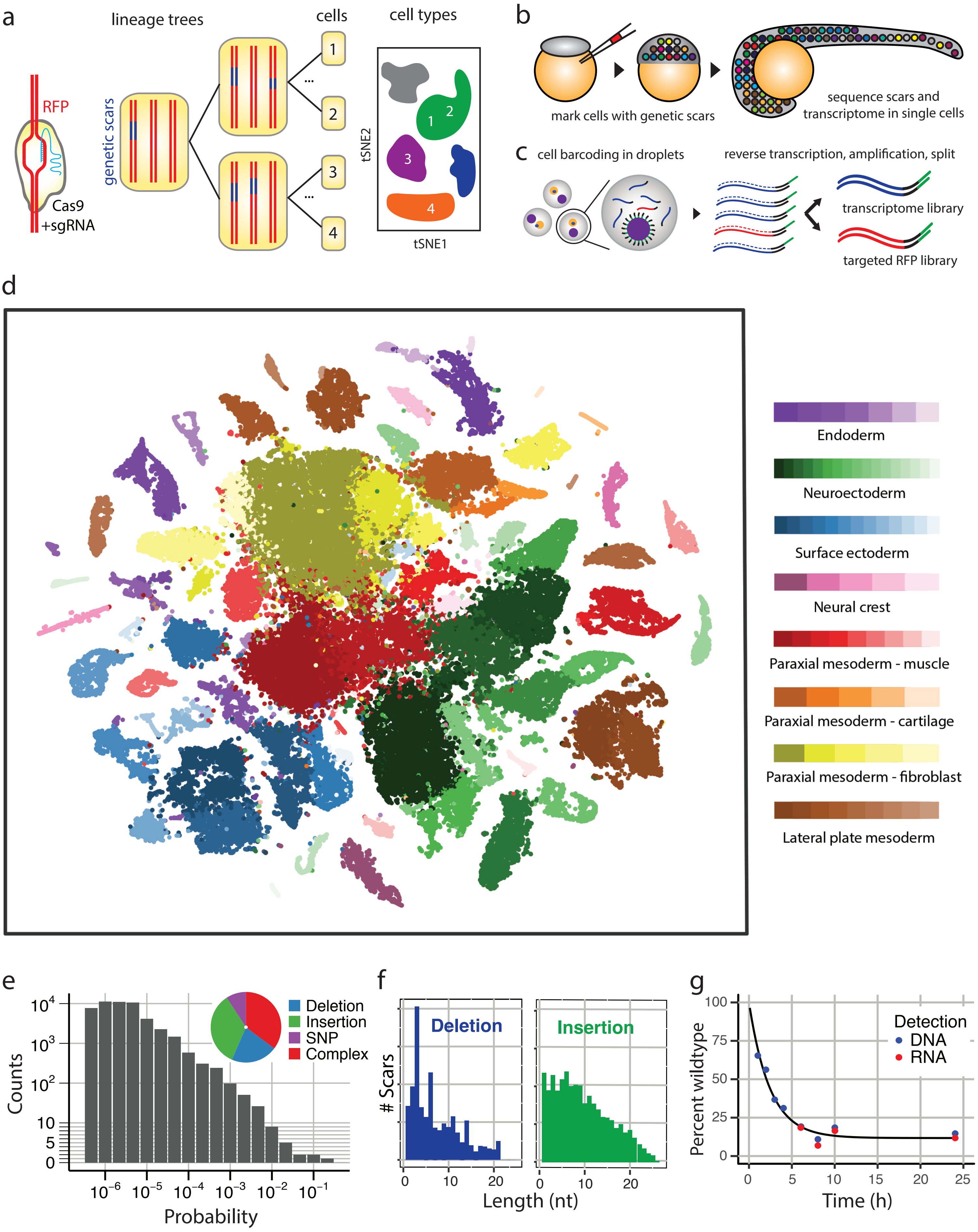
Code availability

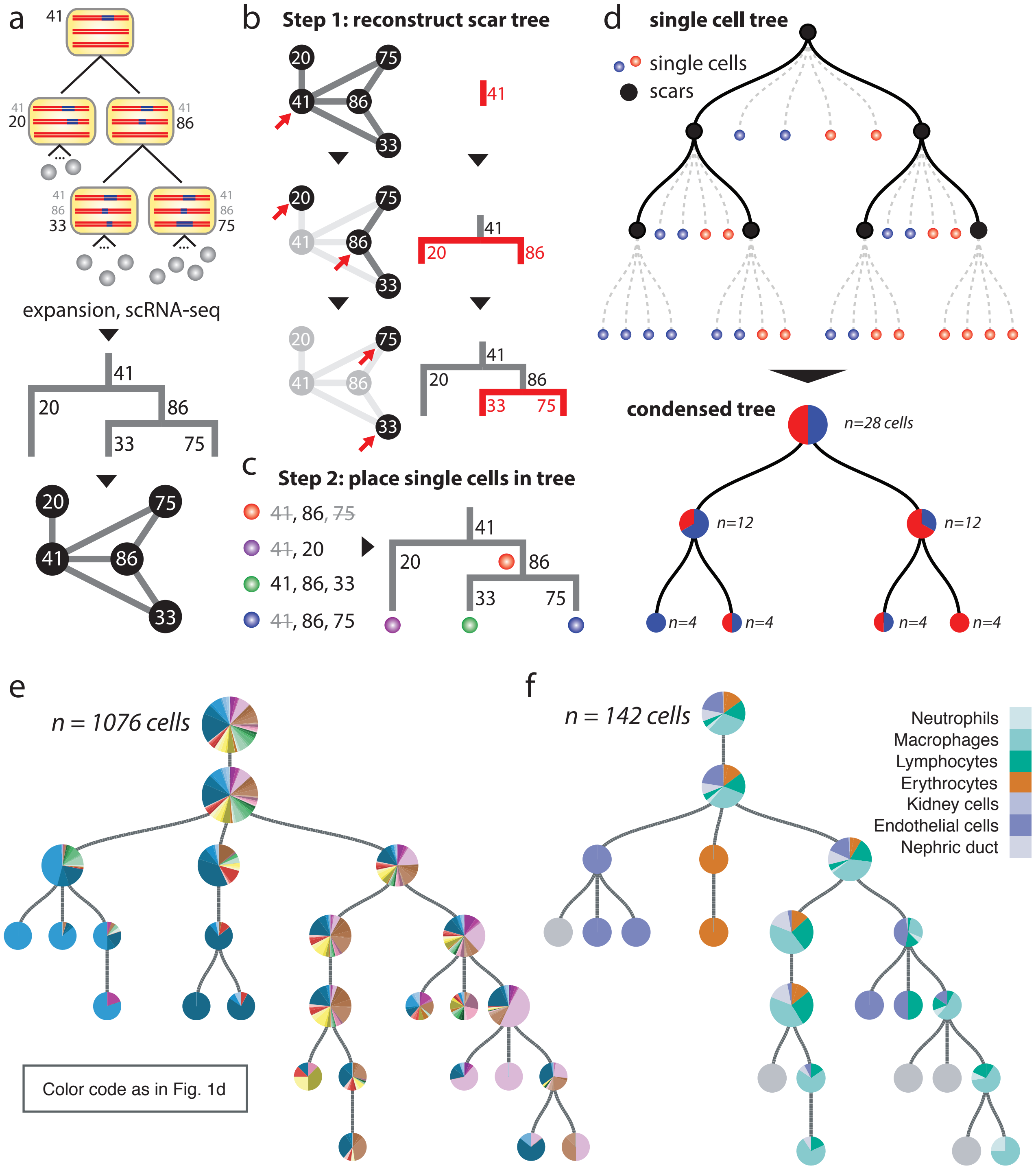
Custom code is provided at https://bitbucket.org/Bastiaanspanjaard/linnaeus. As part of the software package we provide a sample dataset on which the code can be run. The function of all scripts is summarized in README.md.

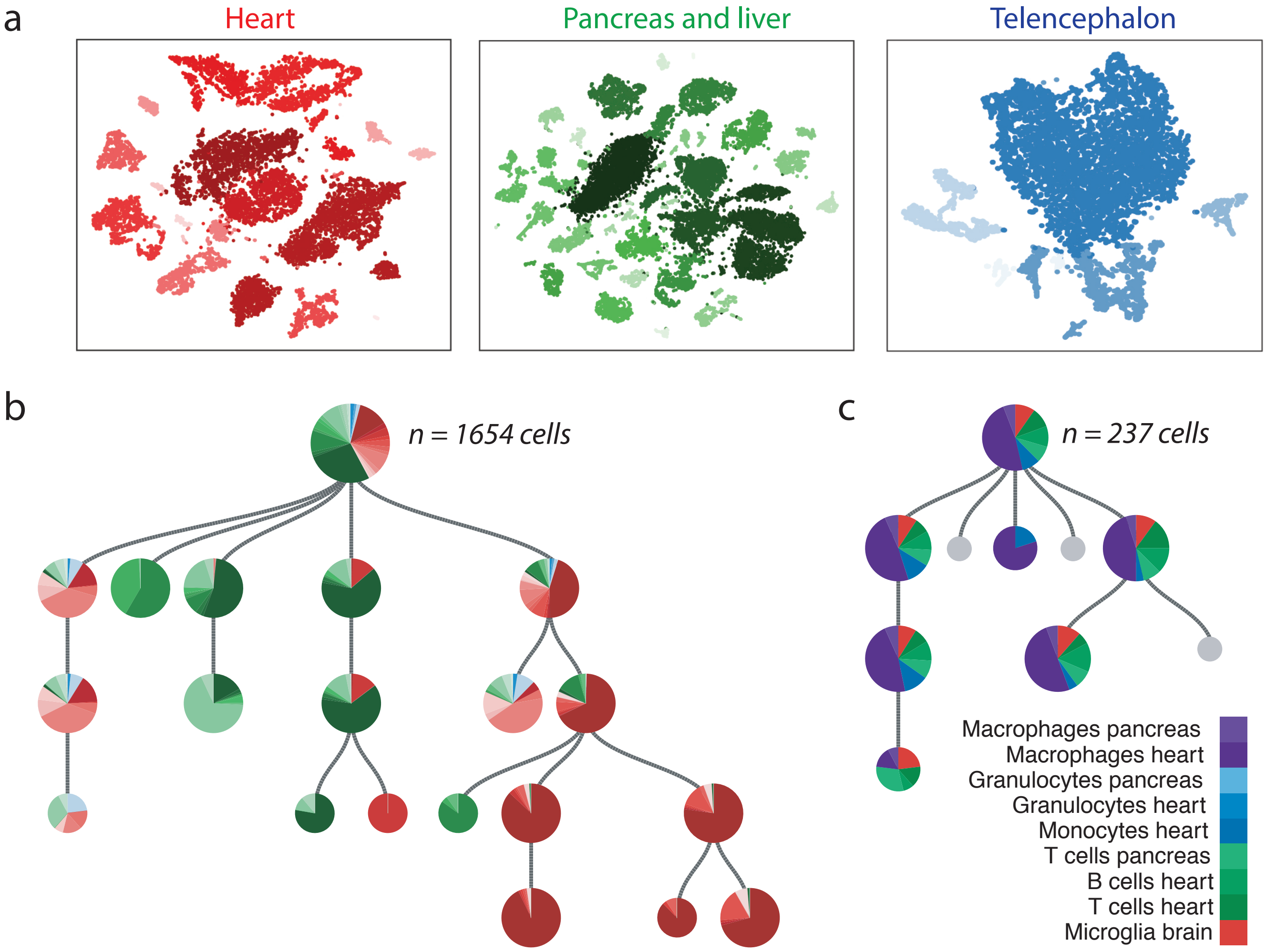
A **Life Sciences Reporting Summary** is available.

Methods-only References

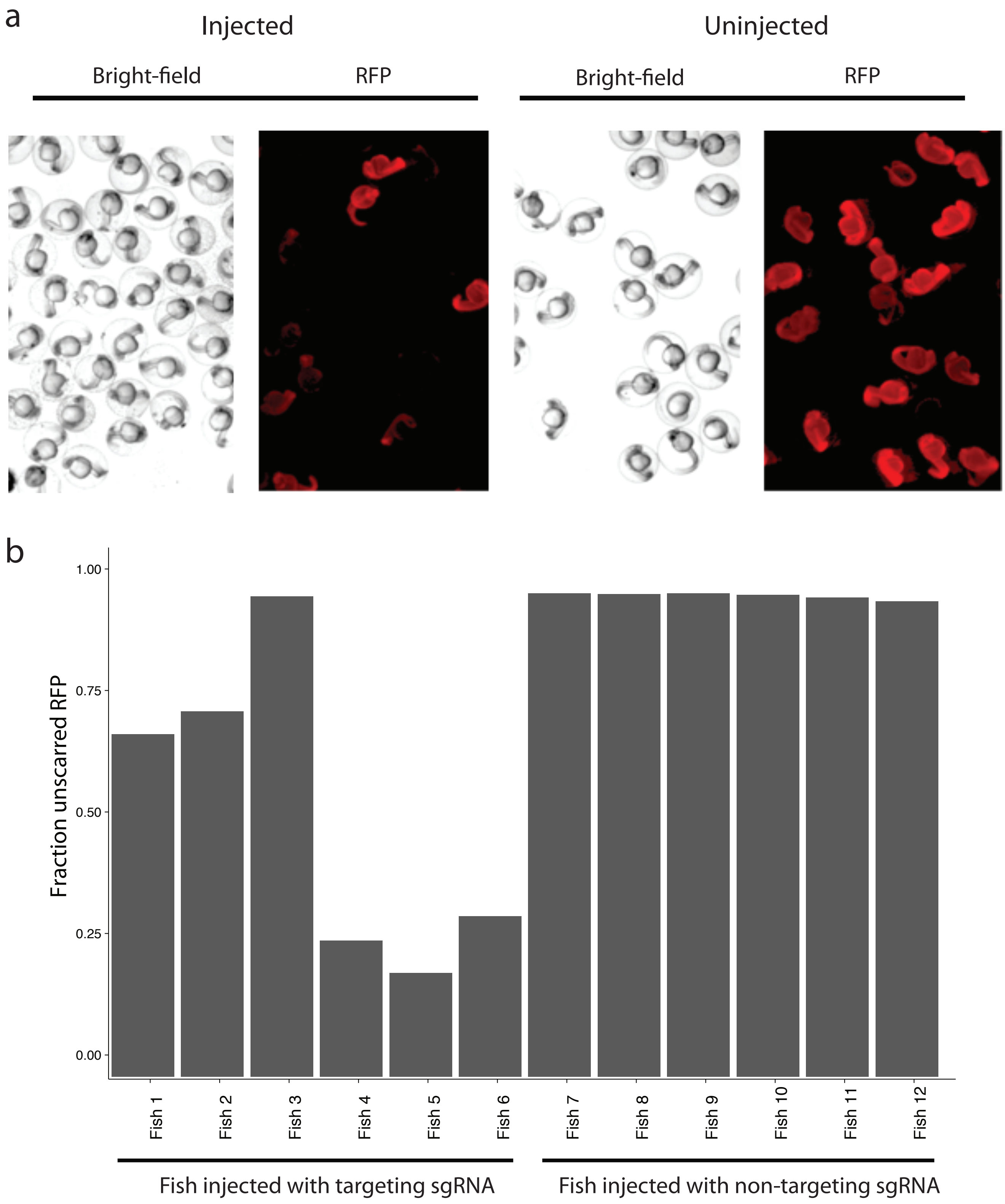
31. Kivioja, T. *et al.* Counting absolute numbers of molecules using unique molecular identifiers. *Nat Methods* **9,** 72–74 (2011).
32. Li, H. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. arXiv:1303.3997v2 (2013).
33. Wang, G. C. & Wang, Y. Frequency of formation of chimeric molecules as a consequence of PCR coamplification of 16S rRNA genes from mixed bacterial genomes. *Appl. Environ. Microbiol.* **63,** 4645–4650 (1997).
34. Waltman, L. & van Eck, N. J. A smart local moving algorithm for large-scale modularity-based community detection. *European Physical Journal B* **86,** (2013).
35. van der Maaten, L. & Hinton, G. Visualizing Data using t-SNE. *Journal of Machine Learning Research* **9,** 2579–2605 (2008).
36. Amir, E.-A. D. *et al.* visNe enables visualization of high dimensional single-cell data and reveals phenotypic heterogeneity of leukemia. *Nature Biotechnology* **31,** 545–552 (2013).
37. McDavid, A. *et al.* Data exploration, quality control and testing in single-cell qPCR-based gene expression experiments. *Bioinformatics* **29,** 461–467 (2013).
38. Howe, D. G. *et al.* ZFIN, the Zebrafish Model Organism Database: increased support for mutants and transgenics. *Nucleic Acids Research* **41,** D854–60 (2013).
39. Kane, D. A. & Kimmel, C. B. The zebrafish midblastula transition. *Development* **119,** 447–456 (1993).
40. Kobitski, A. Y. *et al.* An ensemble-averaged, cell density-based digital model of zebrafish embryo development derived from light-sheet microscopy data with single-cell resolution. *Sci. Rep.* **5,** 8601–10 (2015).
41. Felsenstein, J. 2005. PHYLIP (Phylogeny Inference Package) version 3.6. *Distributed by the author. Department of Genome Sciences, University of Washington, Seattle.*


a
lineage trees
cells
cell types
RFP
Cas9
+sgRNA
genetic scars
1
2
3
4
...
tSNE2
tSNE1
b
mark cells with genetic scars
sequence scars and
transcriptome in single cells
c
cell barcoding in droplets
reverse transcription, amplification, split
transcriptome library
targeted RFP library
d
Endoderm
Neuroectoderm
Surface ectoderm
Neural crest
Paraxial mesoderm - muscle
Paraxial mesoderm - cartilage
Paraxial mesoderm - fibroblast
Lateral plate mesoderm
e
Counts
10^4
10^3
10^2
10
5
0
10^{-6}
10^{-5}
10^{-4}
10^{-3}
10^{-2}
10^{-1}
Probability
Deletion
Insertion
SNP
Complex
f
# Scars
Deletion
Insertion
0
10
20
Length (nt)
g
Percent wildtype
100
75
50
25
0
5
10
15
20
25
Time (h)
Detection
DNA
RNA


a
41
20
86
33
75
expansion, scRNA-seq
b
Step 1: reconstruct scar tree
20
75
41
86
33
c
Step 2: place single cells in tree
~~41~~, 86, ~~75~~
~~41~~, 20
41, 86, 33
~~41~~, 86, 75
d
single cell tree
single cells
scars
condensed tree
n=28 cells
n=12
n=4
e
n = 1076 cells
Color code as in Fig. 1d
f
n = 142 cells
Neutrophils
Macrophages
Lymphocytes
Erythrocytes
Kidney cells
Endothelial cells
Nephric duct

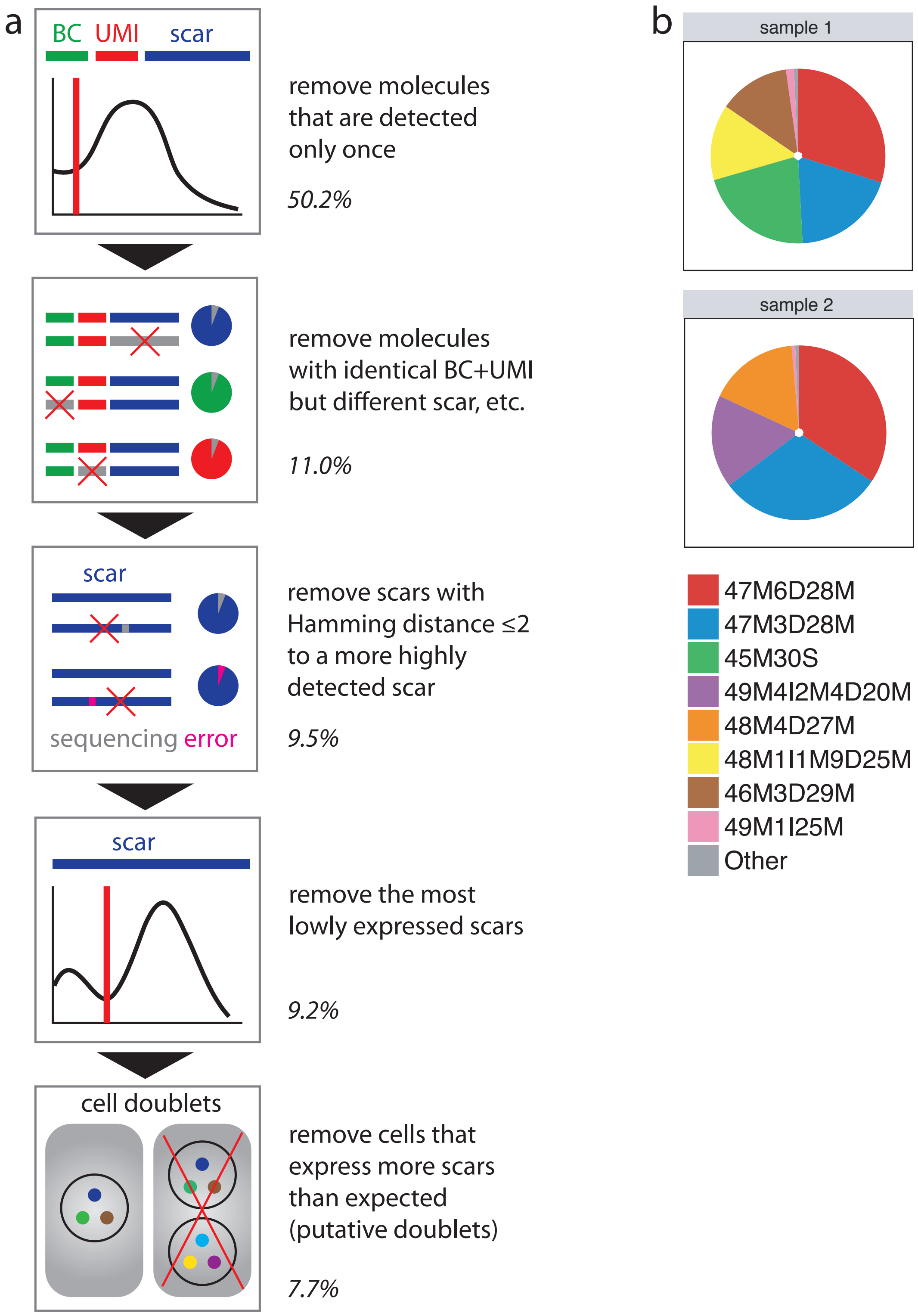

a
Heart
Pancreas and liver
Telencephalon
b
n = 1654 cells
c
n = 237 cells
Macrophages pancreas
Macrophages heart
Granulocytes pancreas
Granulocytes heart
Monocytes heart
T cells pancreas
B cells heart
T cells heart
Microglia brain

Supplementary Figure 1



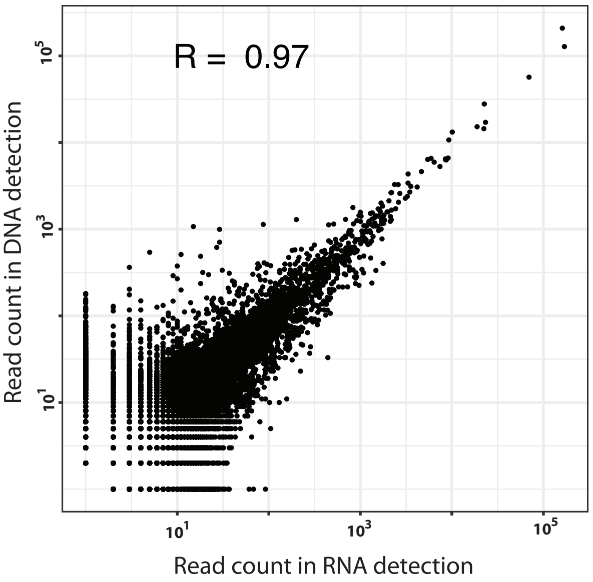
Supplementary Figure 1. Detection of scar formation using microscopy and sequencing (a) Loss of RFP fluorescence upon injection of Cas9 and sgRNA targeting RFP. Experiment was repeated ten times with similar results. **(b)** Reduction of unscarred RFP in sequencing experiment upon injection of Cas9 and sgRNA targeting RFP.

Supplementary Figure 2


a
BC UMI scar
remove molecules that are detected only once
50.2%
remove molecules with identical BC+UMI but different scar, etc.
11.0%
scar
sequencing error
remove scars with Hamming distance ≤2 to a more highly detected scar
9.5%
scar
remove the most lowly expressed scars
9.2%
cell doublets
remove cells that express more scars than expected (putative doublets)
7.7%
b
sample 1
sample 2
47M6D28M
47M3D28M
45M30S
49M4I2M4D20M
48M4D27M
48M1I1M9D25M
46M3D29M
49M1I25M
Other

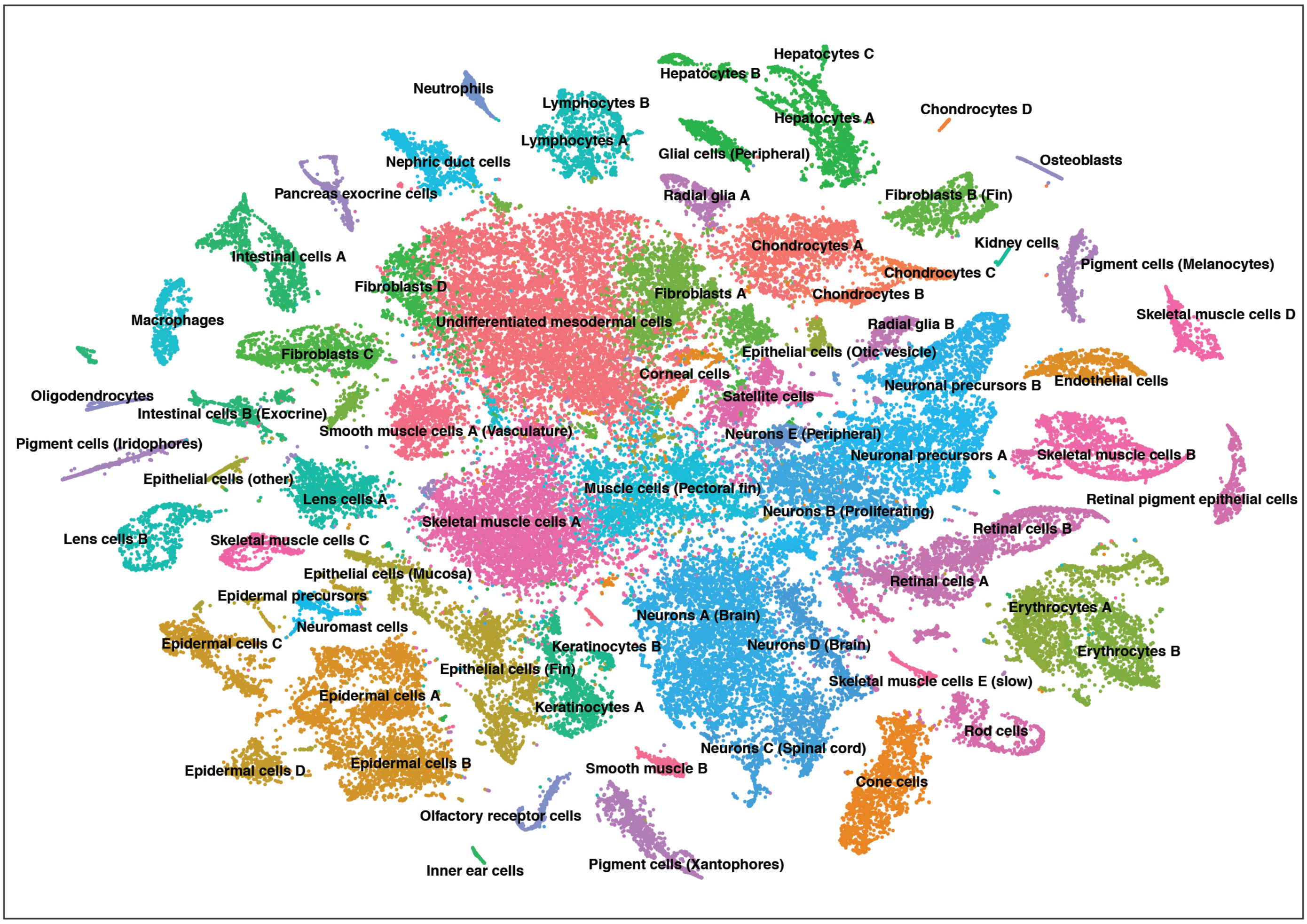
Supplementary Figure 2. Scar filtering pipeline mitigates sequencing errors. To distinguish sequencing errors in scars and barcodes from genuine lineage information, scar libraries are oversequenced and filtered. **(a)** Sketch of scar filtering pipeline. The percentages of correctly mapped reads that are retained after each filtering step is indicated for a selected dataset. For a detailed description of the individual filtering steps see Online Methods. **(b)** Validation of the filtering pipeline with two control datasets in which single cells from the offspring of Cas9 injected parents were analyzed. In these fish, all cells have the same scar pattern, leading to a strong reduction of scar diversity. Detected scar sequences are labeled by their CIGAR code. We confirmed by DNA-based bulk sequencing of siblings of the two fish that 49M1I25M is indeed a genuine scar (data not shown). The category “Other”, corresponding to putative sequencing errors that may have escaped detection in our scar filtering pipeline, constitutes less than 1% of the scars. Spurious scar connections created by e.g. sequencing errors are not considered for lineage tree reconstruction (see Online Methods). In Supplementary Fig. 7, this dataset is used to analyze scar detection efficiency in a cell type dependent manner.

Supplementary Figure 3



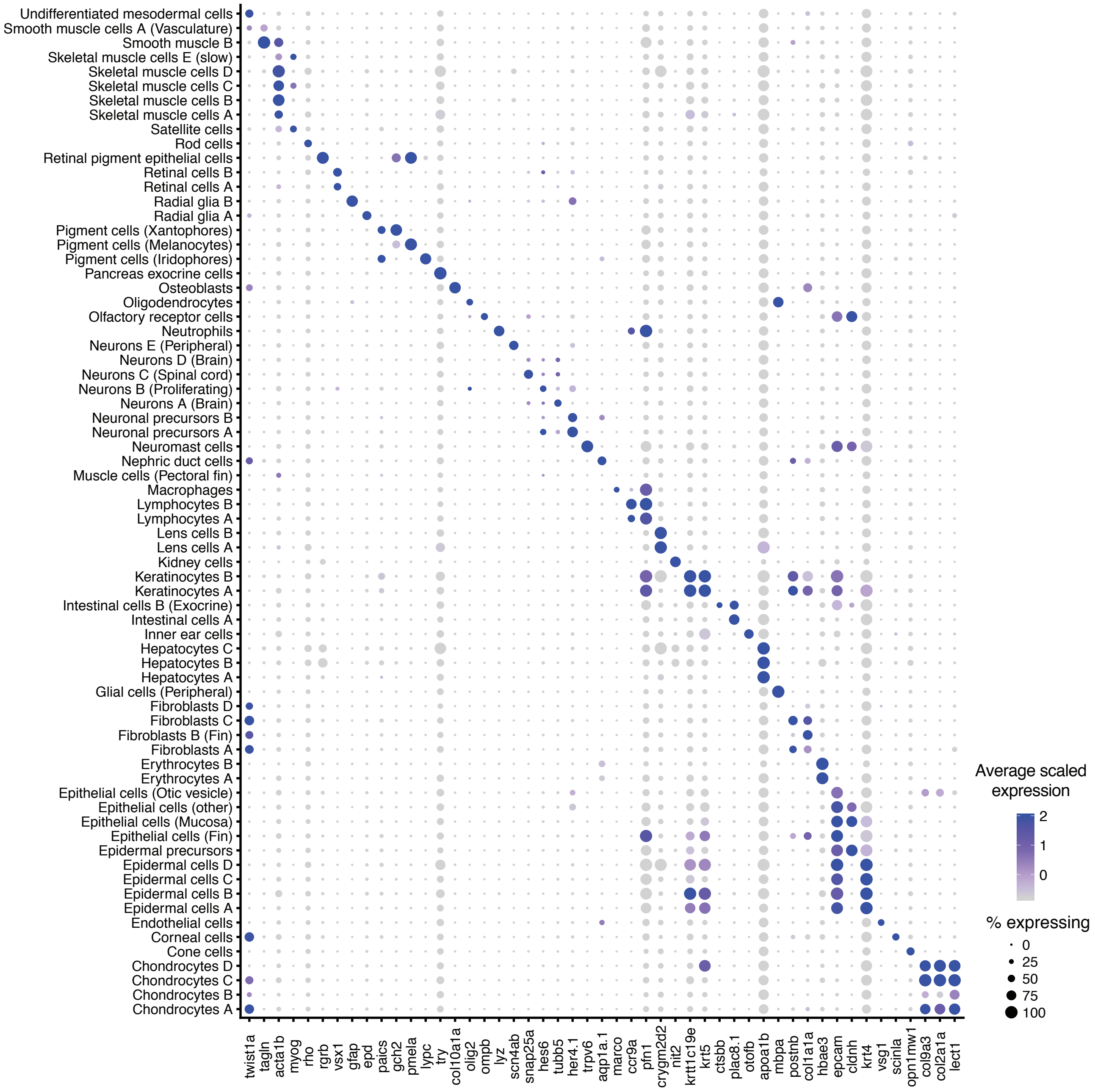
Supplementary Figure 3. Strong correlation between DNA and RNA detection in the same embryo. Scar abundances detected in bulk on the DNA and RNA level for 24 hpf zebrafish larvae (70,251 scar sequences from three animals) are strongly correlated (Pearson correlation coefficient 0.91).

Supplementary Figure 4



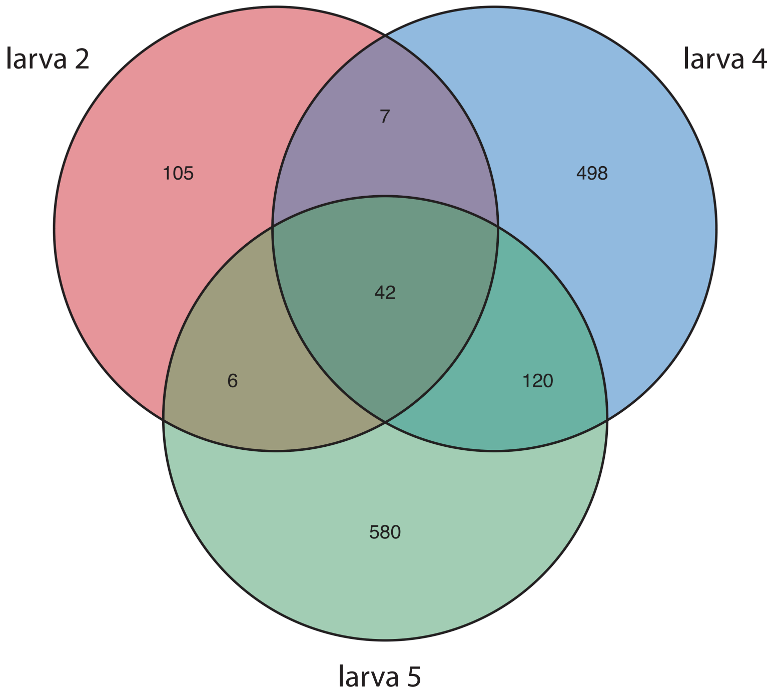
Supplementary Figure 4. Cell type identification in 5 dpf zebrafish larvae (n=7 animals). tSNE representation of clustering results for scRNA-seq data. Labels indicate cell type assignment.

Supplementary Figure 5



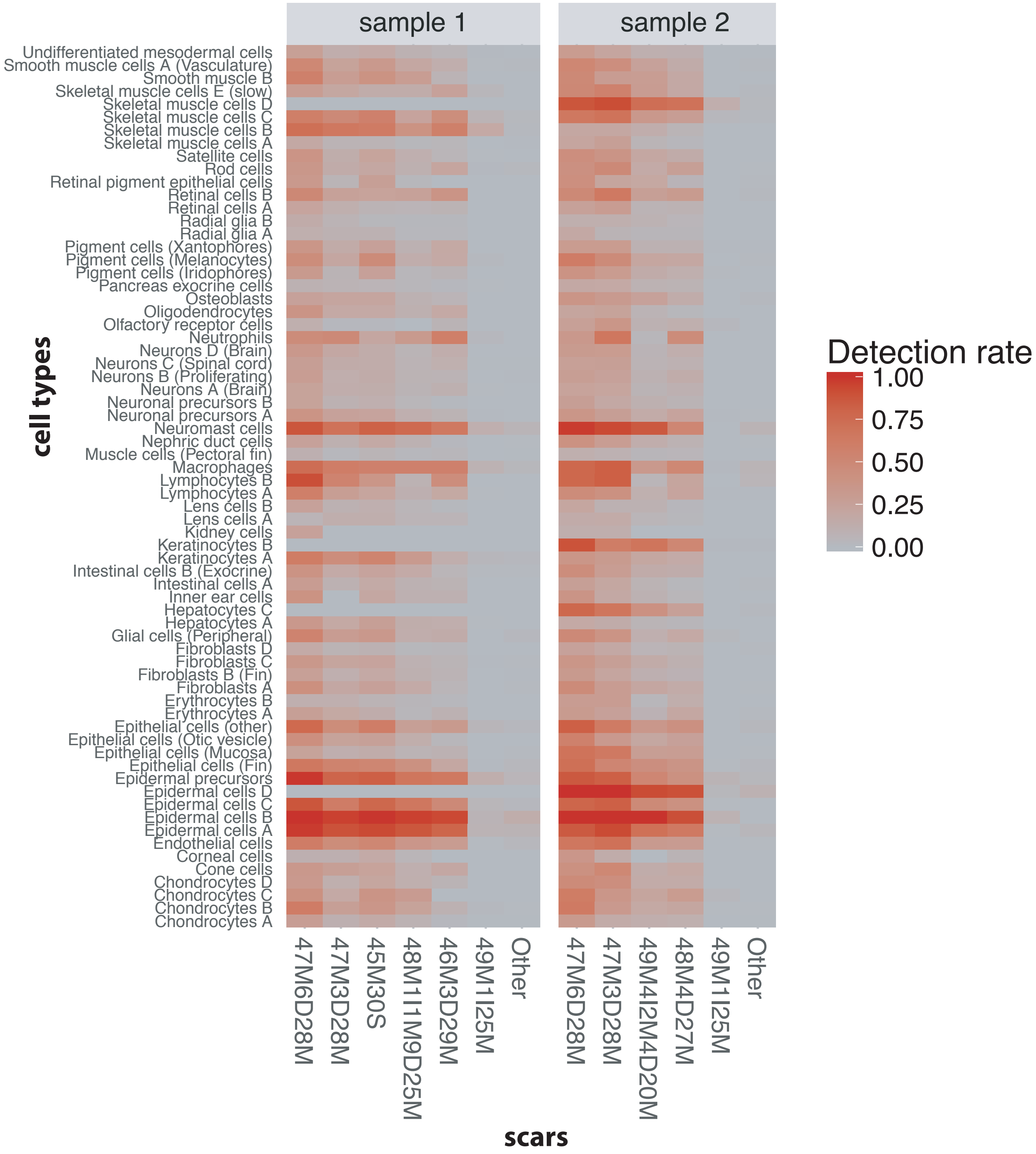
Supplementary Figure 5. Cell type markers in 5 dpf zebrafish larvae. Gene expression patterns for selected markers across all clusters. Dot size represents the percentage of cells expressing the marker; color represents the average scaled expression level.

Supplementary Figure 6



Supplementary Figure 6. Scar diversity. Venn diagrams indicating numbers of unique and overlapping scars in three 5 dpf larvae (larva 2 corresponds to the tree shown in Fig. 2d).

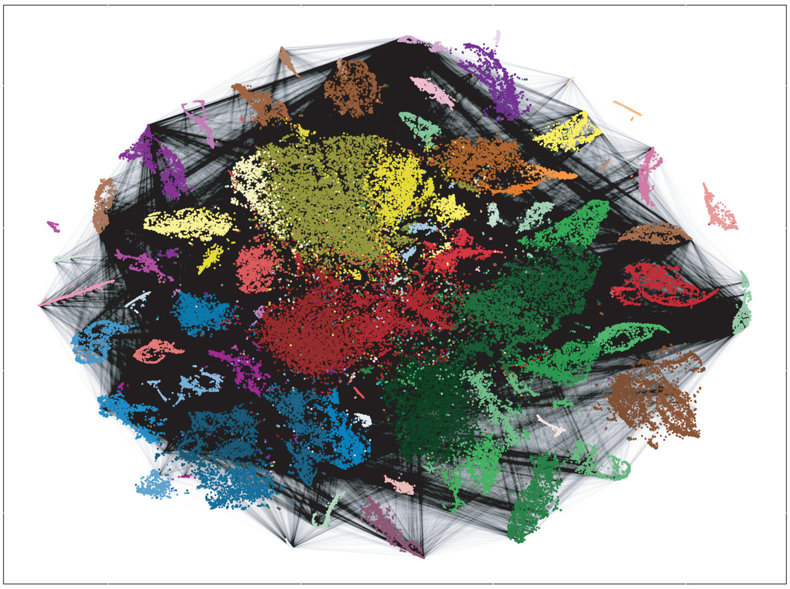
Supplementary Figure 7



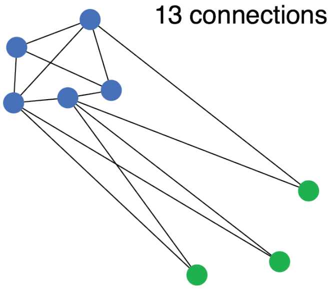
Supplementary Figure 7. Single-cell scar detection rates. To measure the cell type dependence and integration site dependence of scar detection efficiencies, we analyzed the offspring of Cas9 injected fish on the single cell level. We found that scar detection efficiency depends on cell type, which probably reflects differences in cell size or promoter strength. While most genomic integrations (i.e. different scar sequences) have similar detection efficiencies, we also found one integration (CIGAR code 49M1I25M) with significantly lower expression. Such lowly expressed scars are not considered for tree reconstruction (see Online Methods), since they carry very little lineage information and may be hard to distinguish from sequencing errors (category “Other”, see also Supplementary Fig. 2). Importantly, we did not find any highly expressed scars that were undetectable in specific cell types, suggesting that developmental silencing of specific integrations is no major concern. We used these results to simulate lineage tree reconstruction under realistic conditions (Supplementary Fig. 11).

Supplementary Figure 8

a



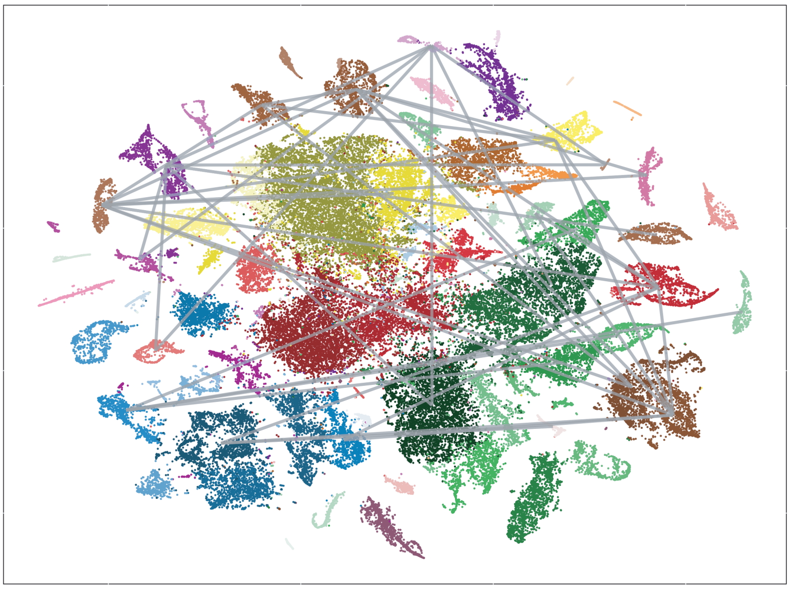
b



Cluster	Connection endpoints	Chance of picking endpoint
Blue	20	20/26
Green	6	6/26

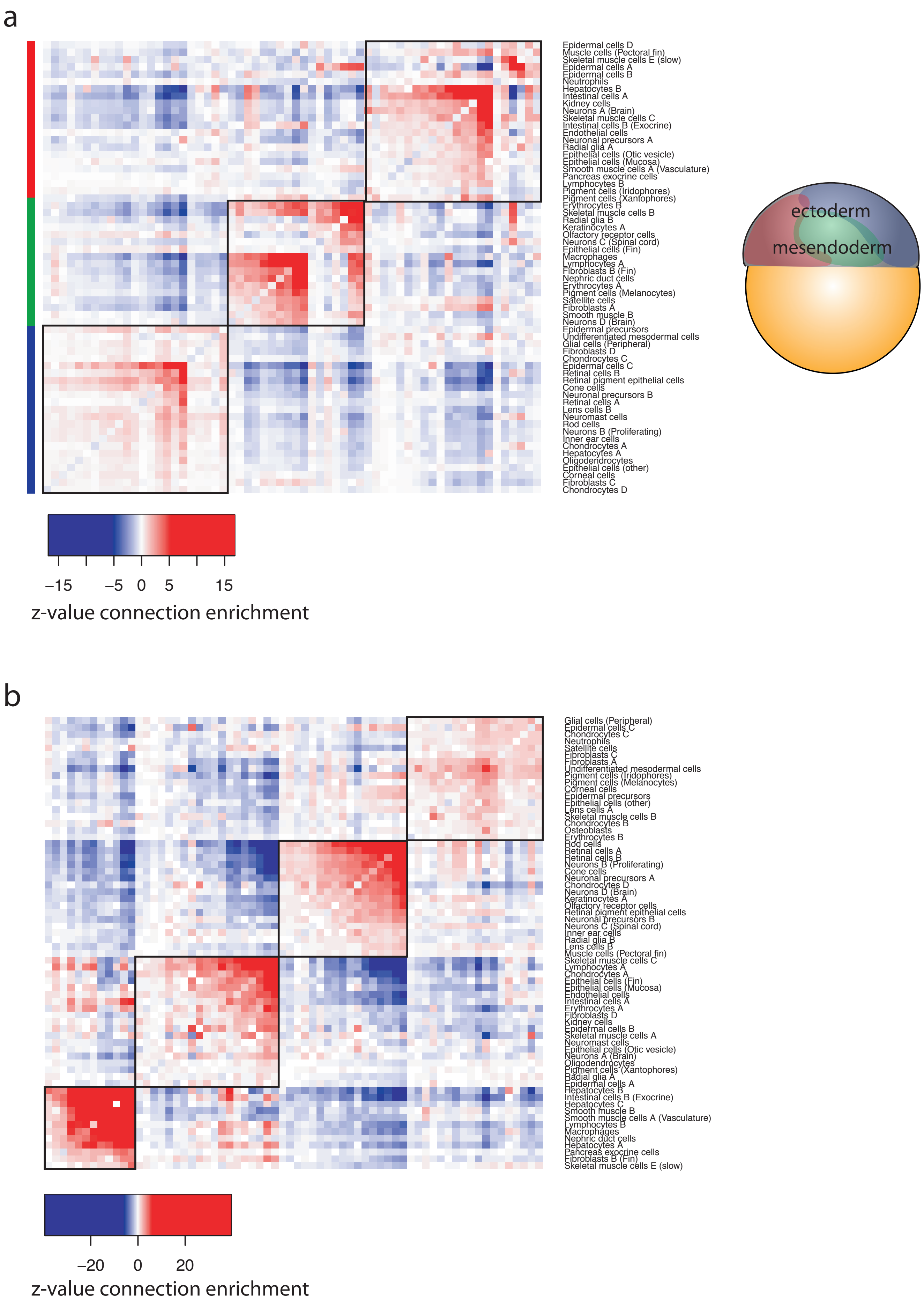
Connection	Chance
Blue-blue	0.592
Green-green	0.053
Blue-green	0.355

c



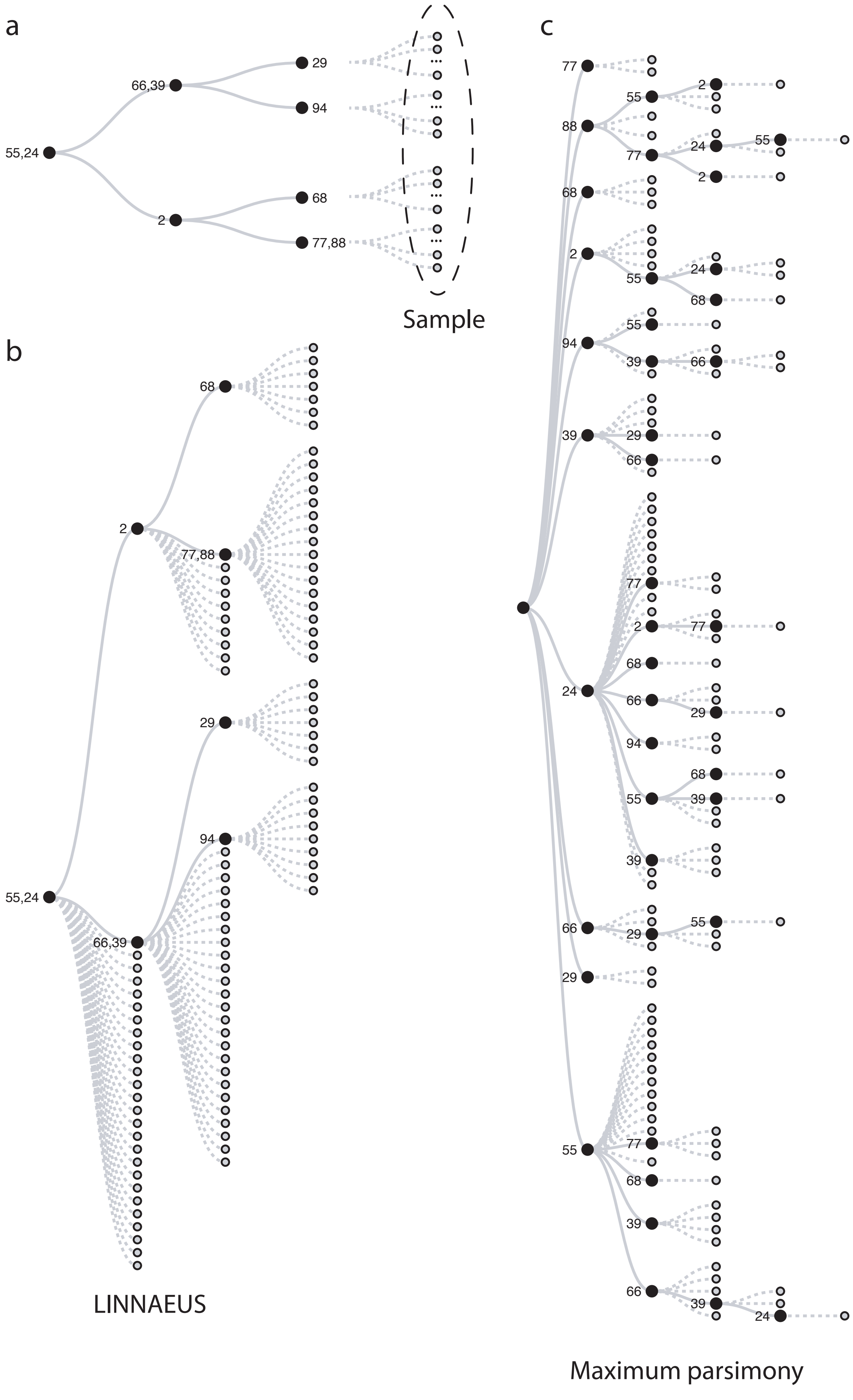
Supplementary Figure 8. Enriched connections between pairs of cell types. (a) t-SNE representation of scRNA-seq data for dissociated zebrafish larvae (5 dpf, n=7 animals). Pairs of cells with at least one common scar are linked by black lines. Scars with creation probability ≥0.001 and scars that were detected in more than 1 larva were excluded from the analysis. **(b)** Example of background model for analysis of enriched connections (see also Online Methods). Given twenty connection endpoints in blue cells, and six in green cells, the chance of a blue endpoint is 10/13, and the chance of a green endpoint is 3/13. We can then calculate the chances of randomly selecting a blue-blue connection, a blue-green connection and a green-green connection. These chances determine a binomial distribution of connections to compare the observed connections against. **(c)** Enrichments of scar connections between cell types compared to random distributions of n=289,452 connections using a two-sided binomial test followed by Benjamini-Hochberg correction. Only enrichments with p_{adj}<0.01 are shown.

Supplementary Figure 9



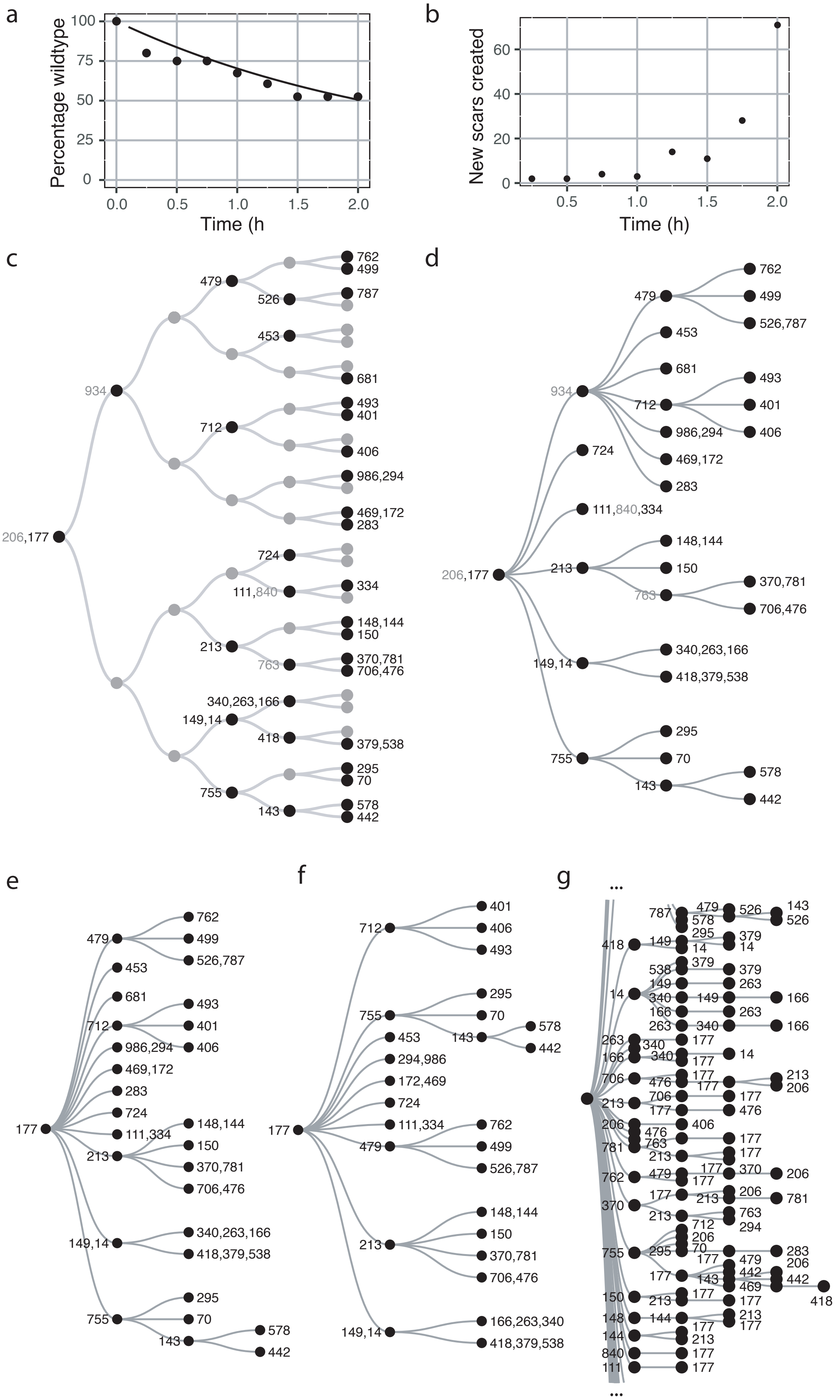
Supplementary Figure 9. Enriched connections form clusters on fate map. (a) Hierarchical clustering of cell types by scar connection strength yields three clusters. We calculated z-values of connection enrichments between cell types using an underlying binomial distribution (see Online Methods). Cell type distances were calculated based on these connection enrichments, and the cell types were clustered on these distances. Cartoon on the right: Cell type clusters determined based on scar connections form contiguous domains on the zebrafish fate map at shield stage (yolk sac shown in orange). This sketch is an oversimplification since it does not show the other hemisphere of the embryo (which may have different cluster boundaries) and since scar clones do not expand strictly locally (there is some degree of mixing of cells). **(b)** Segregation of cell types by scar connection strength is reproducible in another larva. Here we detect four instead of three clusters of cell types, which is probably linked to stochastic fluctuations in the number of scars that are created during the first few cell divisions.

Supplementary Figure 10



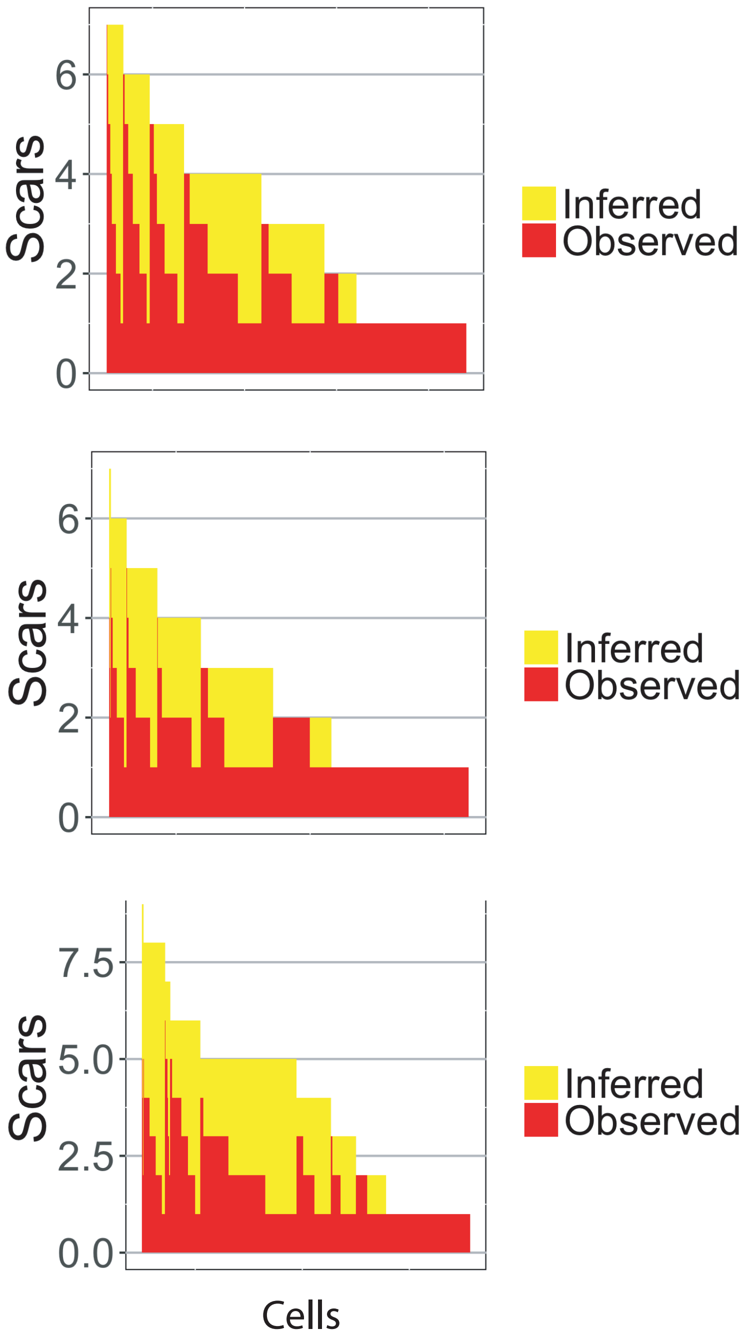
Supplementary Figure 10. Comparison of LINNAEUS tree building to maximum parsimony on simple model tree. (a) Simulation of a developmental lineage tree with two cell divisions. The numbers next to a precursor cell (black node) denote the scars created in that cell. We expand the four final branches without further scarring and sample 99 cells (dashed lines, gray nodes) with scar detection rate 0.3. **(b)** The LINNAEUS tree building algorithm exactly recreates the developmental lineage tree of scarring events in precursor cells (solid lines, black nodes). Each sampled cell is placed at the lowest scar node possible according to the scars observed in that cell (dashed lines, gray nodes). **(c)** Maximum parsimony tree (with all scars not observed coded as “not present”) does not recreate the developmental lineage tree faithfully but generates unjustified complexity including multiple creation of the same scar sequences.

Supplementary Figure 11



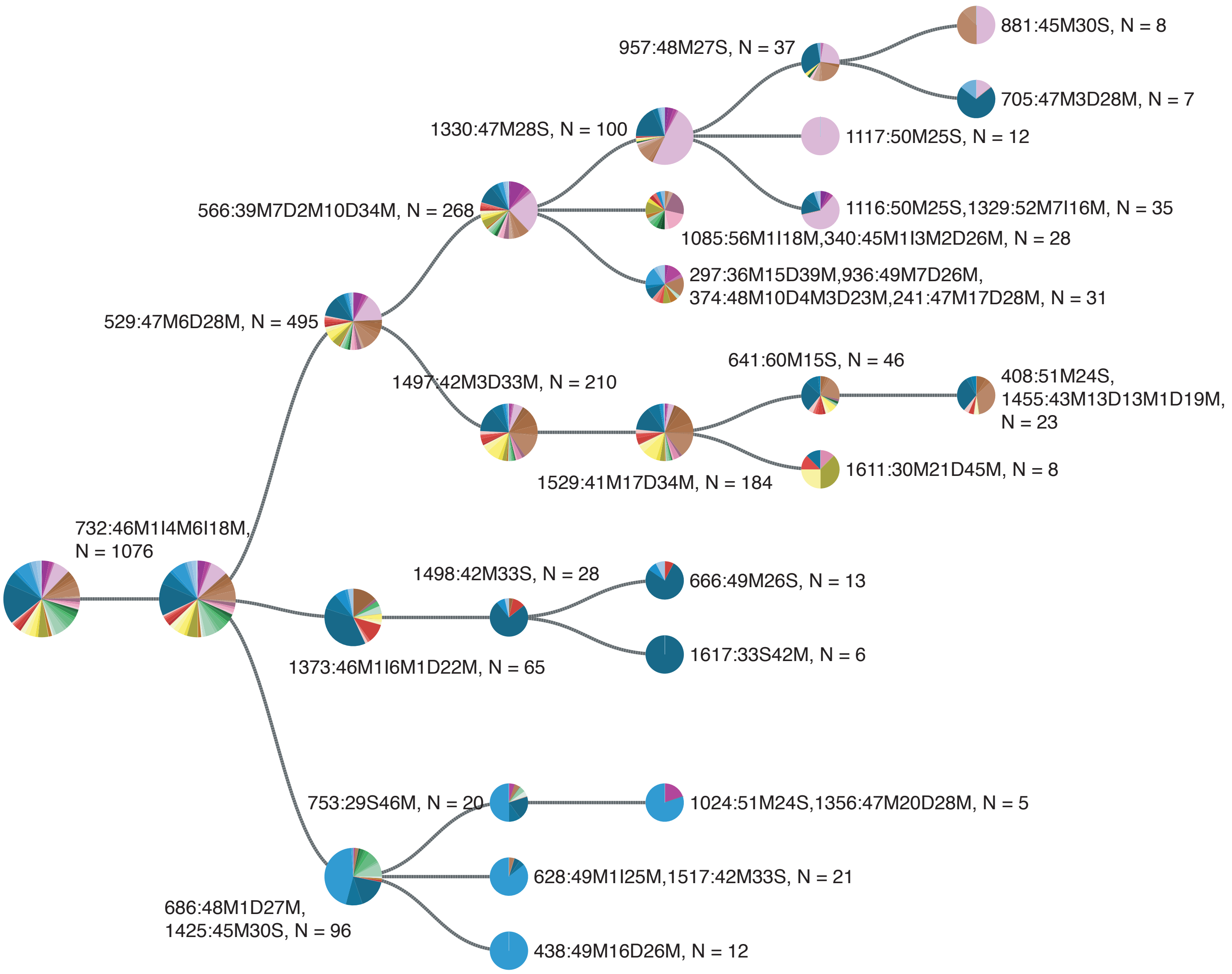
Supplementary Figure 11. Comparison of LINNAEUS tree building to maximum parsimony on a realistic experiment. (a) Using cell division rates as determined by lightsheet microscopy (Kobitski et al., 2015), a scarring rate of 0.4 scars per hour reproduces the experimental results well. Black line shows the exponential fit from Fig. 1g, dots show fraction wildtype at the individual steps of the simulation. **(b)** Number of new scars per cell division according to the simulation. Initially, the number of new scars increases over time, due the cell division rate being higher than the scarring rate. The number of new scars per unit time will eventually drop as the cell division rate slows down, Cas9 concentration decreases, and available target sites are used up. **(c)** Simulation of a developmental lineage tree with six cell divisions and ten target sites, assuming three cell types (fraction 15%, 25%, 60%) with different detection rates (70%, 30%, 15%, respectively). We furthermore assumed that two of the ten target sites are much harder to detect (by a factor 20, i.e. detection rates 3.5%, 1.5%, 0.5%). Due to stochasticity, scars are not created in all precursor cells. Precursor cells in which a scar is created are black, precursor cells in which no scar is created are gray. Scars in weakly expressing sites are also shown in gray. We expand all final branches (not shown) and sample 1716 cells from the resulting pool with a cell doublet rate of 5%. **(d)** The scar tree: the lineage tree that can be measured by scars. Gray nodes in (c) are collapsed to retain only those precursor states that can be observed. **(e)** Simulated scar tree without scars from weakly expressing sites. This is the tree that should be identified by our algorithm, since lowly expressed scars may be difficult to distinguish from sequencing errors and do not carry enough information to justify inclusion in the tree. **(f)** The LINNAEUS tree building algorithm faithfully recreates the scar tree. **(g)** Clipping of maximum parsimony tree (with all scars not observed coded as “not present”). Maximum parsimony does not recreate the developmental lineage tree but introduces considerable amounts of unjustified complexity. Overall, the reconstructed tree contains 46 scars and 265 scarring events, which means each scar is placed on average more than 5 times in the tree.

Supplementary Figure 12



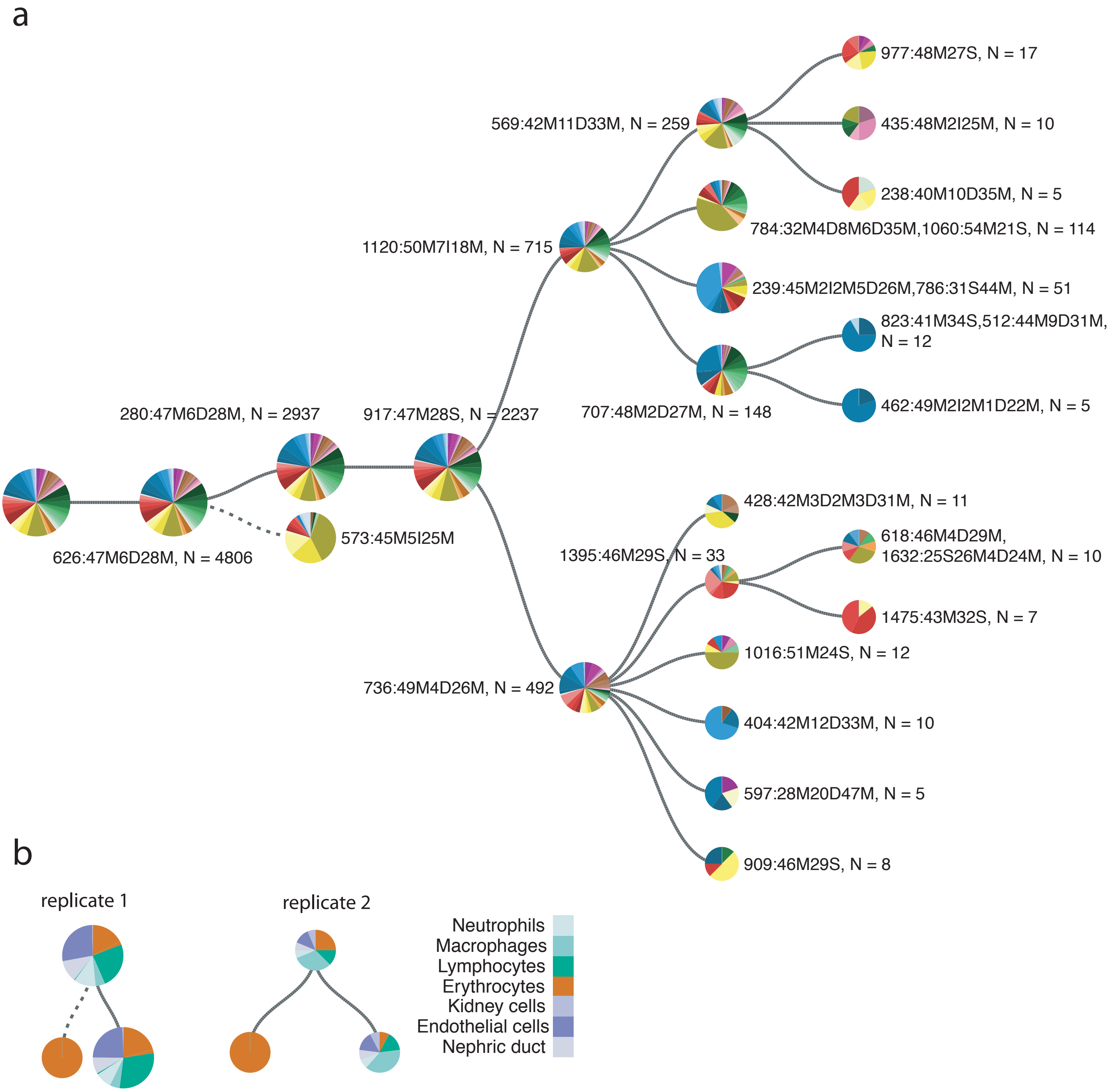
Supplementary Figure 12. Lineage tree reconstruction allows inference of scars that were present but not detected. Detected (red) and inferred (yellow) scars in single cell data from three different 5 dpf larvae. Scar inference is based on the principle that, if the order of scarring is known, detection of a scar allows inference of all previously created scars. For instance, for the blue cell in Fig. 2c, scar 41 can be inferred, since scars 86 and 75, that were created later, were detected.

Supplementary Figure 13



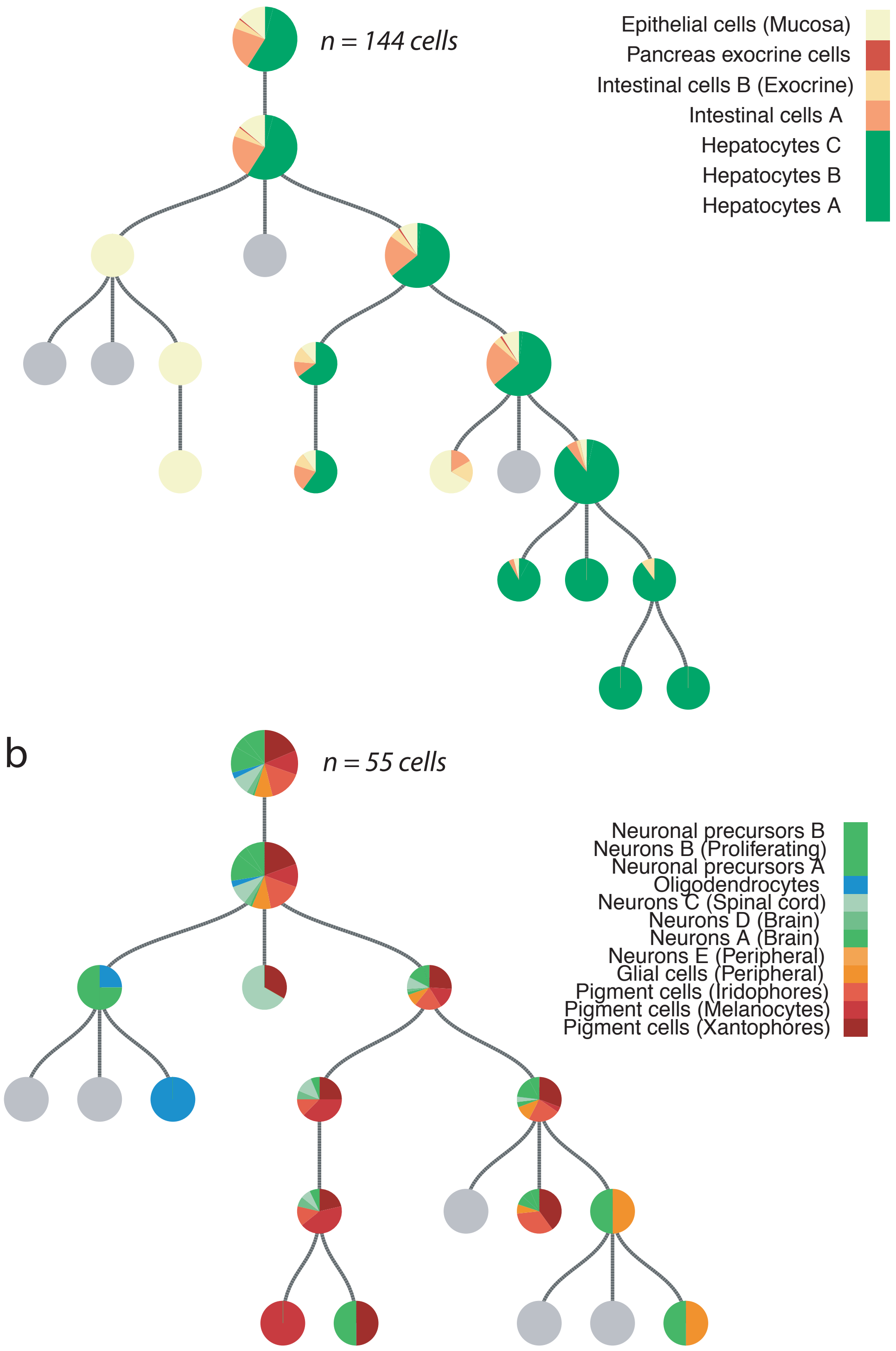
Supplementary Figure 13. Technical information for 5 dpf lineage tree. Scar identifiers (frequency ranking according to Fig. 1e, and CIGAR code) and cell numbers for the lineage tree shown in Fig. 2e. Color code as in Fig. 1d.

Supplementary Figure 14



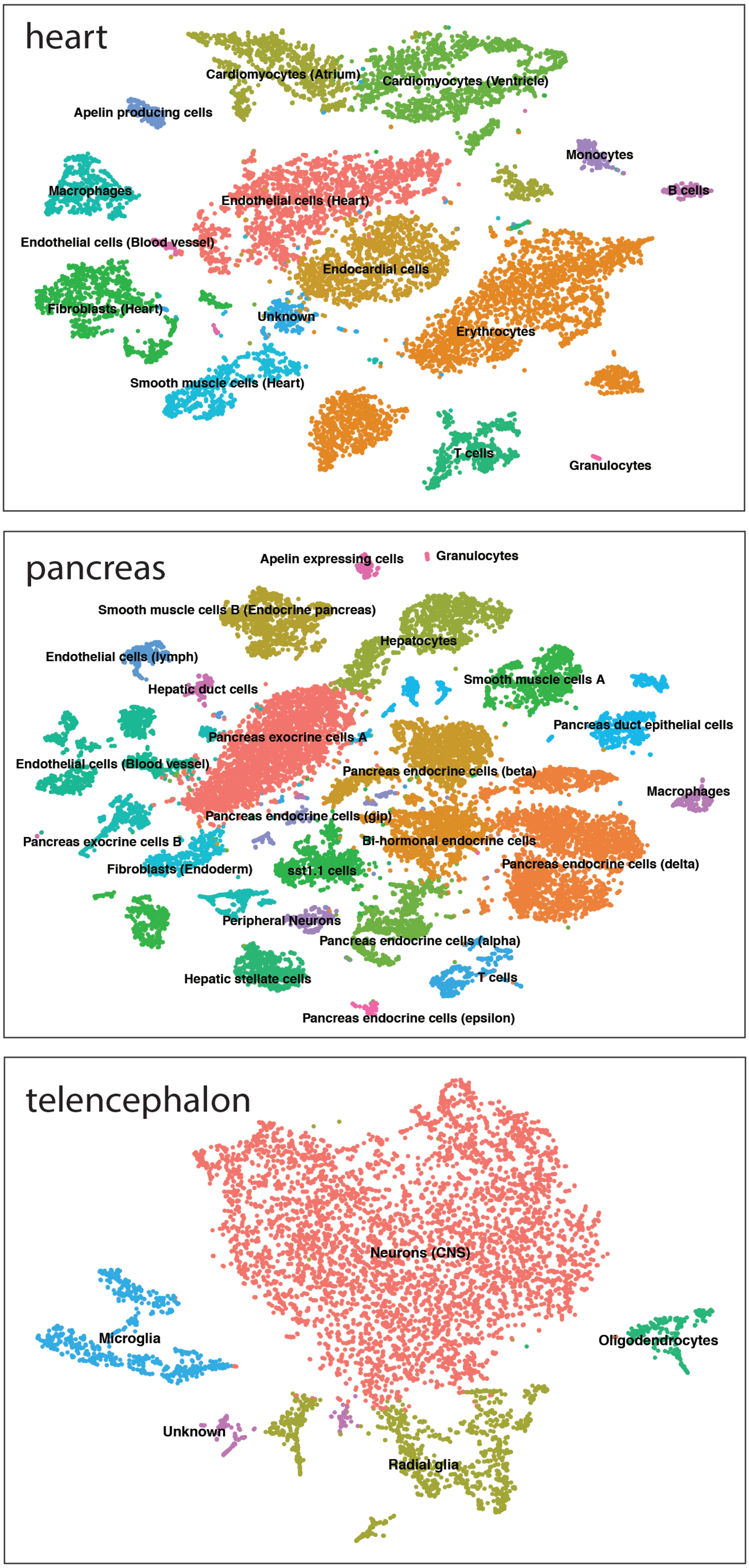
Supplementary Figure 14. Replicate experiments 5 dpf larvae. (a) Full tree for another 5 dpf larva, including scar identifiers (frequency ranking according to Fig. 1e, and CIGAR code) and cell numbers. Color code as in Fig. 1d. **(b)** Mini trees zooming into lateral plate mesoderm for two more 5 dpf larvae.

Supplementary Figure 15



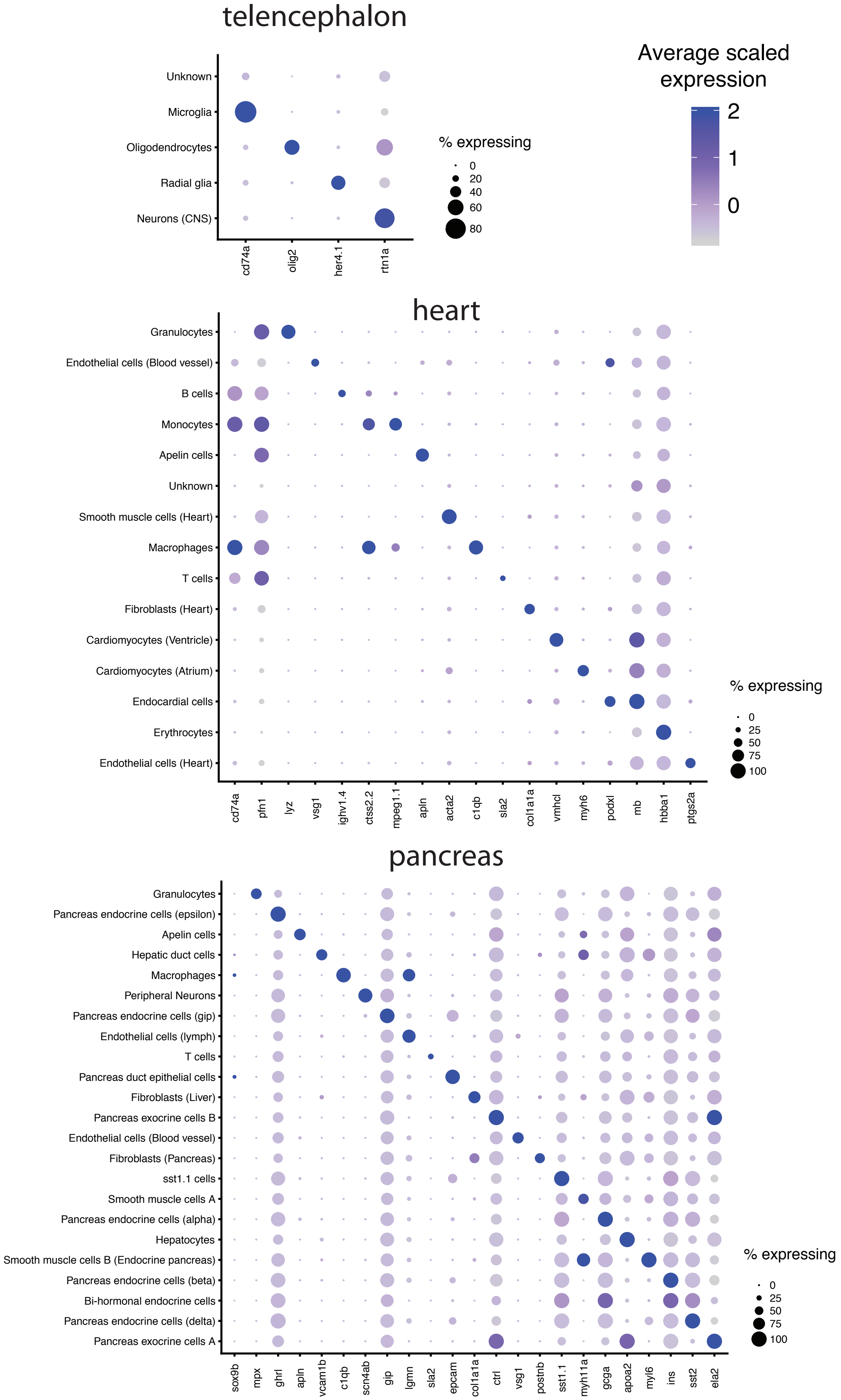
Supplementary Figure 15. Endodermal and neuronal/neural crest lineage trees in 5 dpf larva. Zoom into endodermal **(a)** and neuronal/neural crest **(b)** cell types for the 5 dpf larva analyzed in Fig. 2e. We observe low clonality in the endoderm, with most cell types following the same major lineage branch. The neural crest cell types also have a shared lineage tree, which overlaps with a subset of the neuronal cell types (green in Fig. 2e).

Supplementary Figure 16



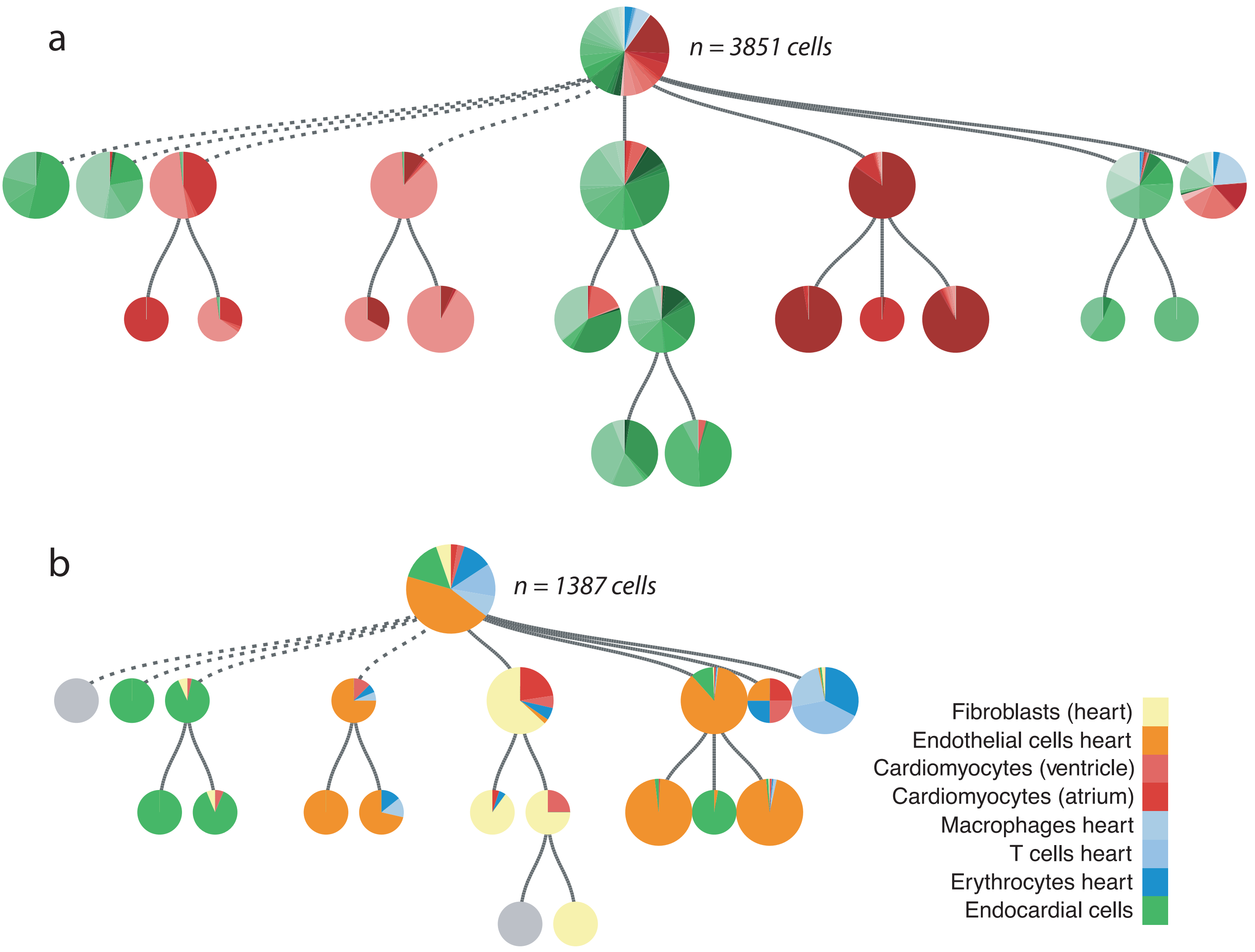
Supplementary Figure 16. Cell type identification in adult zebrafish organs. tSNE representation of clustering results for scRNA-seq data in dissociated heart, pancreas and telecephalon (n=3 animals). Labels indicate cell type assignment.

Supplementary Figure 17



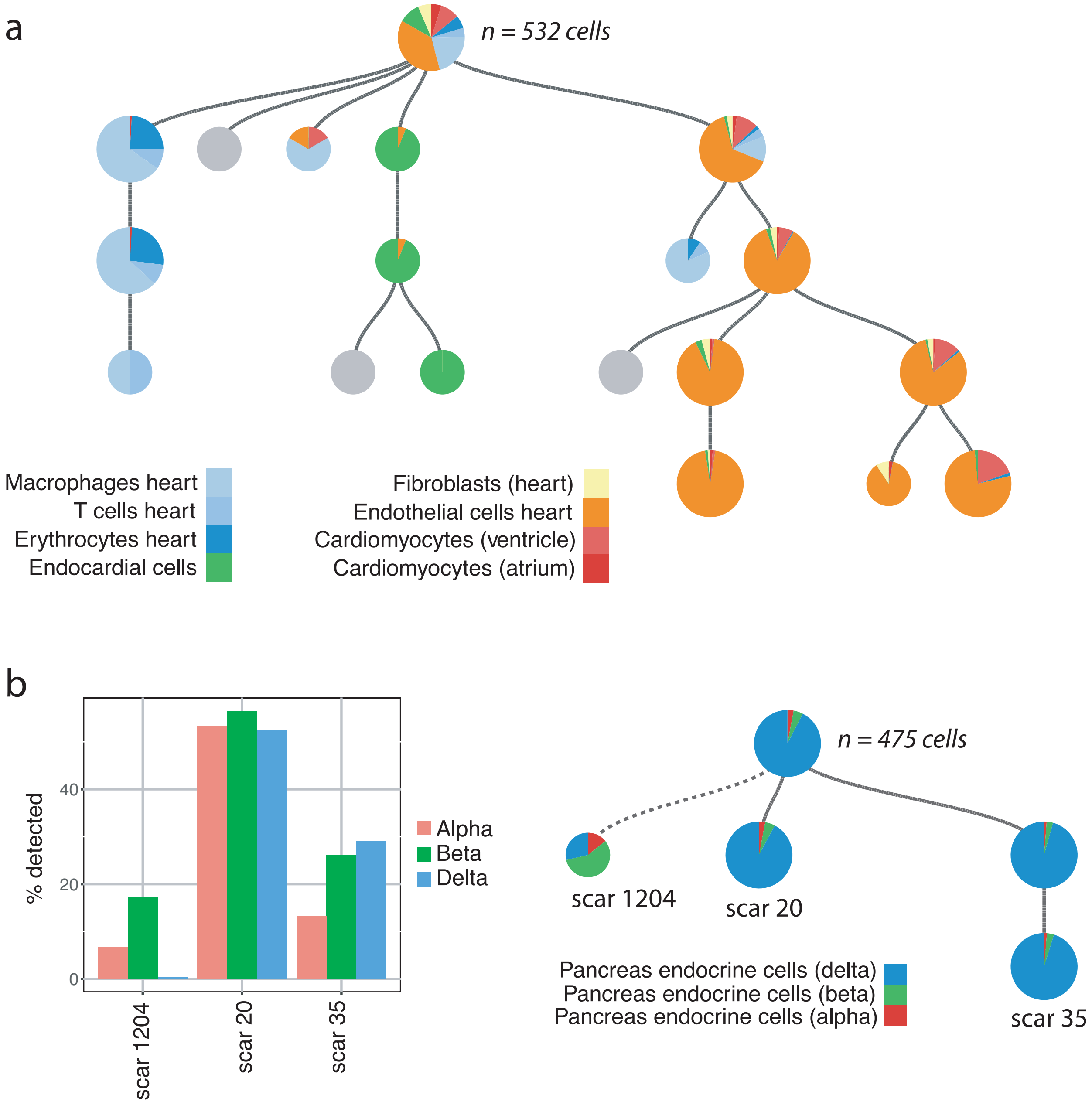
Supplementary Figure 17. Cell type markers in adult zebrafish organs. Gene expression patterns for selected markers across all cell types in dissociated heart, pancreas and telecephalon. Dot size represents the percentage of cells expressing the marker; color represents the average scaled expression level.

Supplementary Figure 18



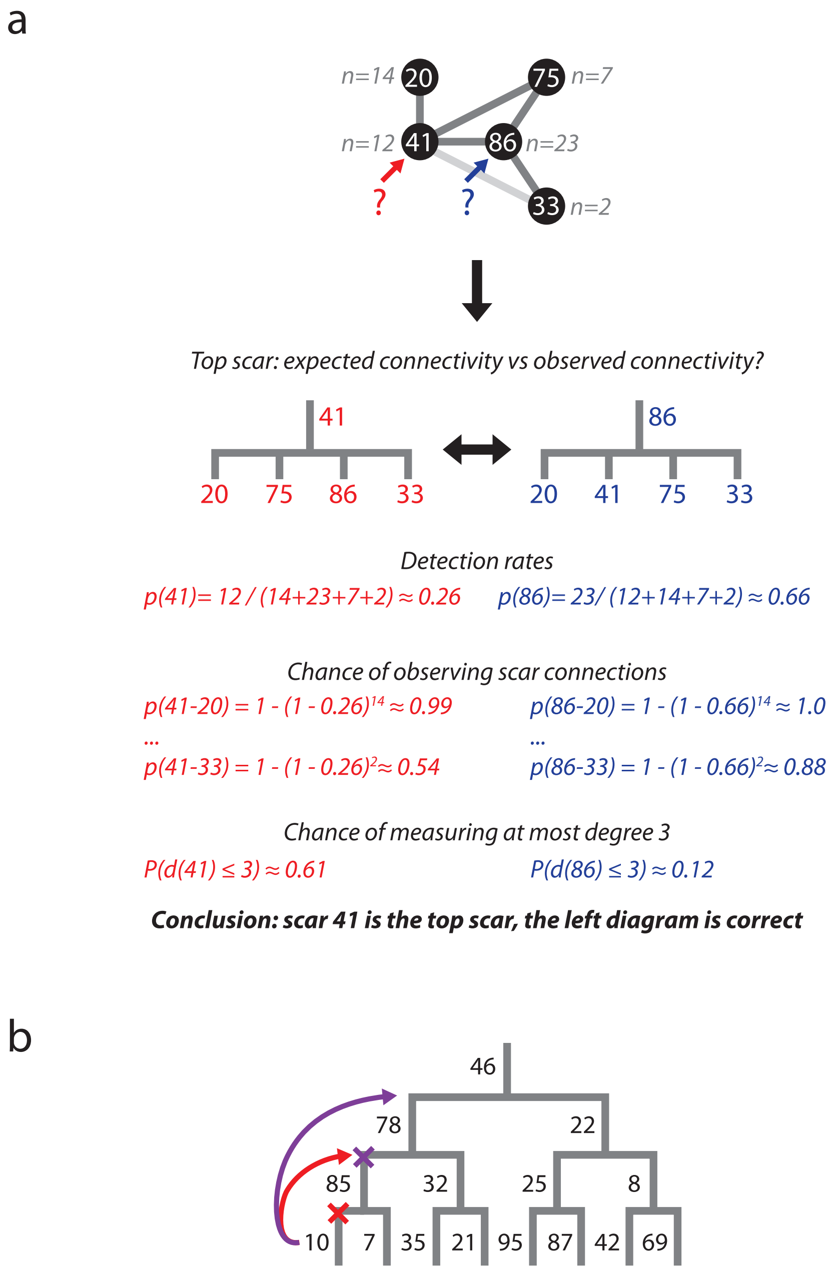
Supplementary Figure 18. Replicate experiment for dissected organs of one adult zebrafish. (a) Full lineage tree. Similar to Fig. 3b, cell types cluster by organ in the lineage tree (with the exception of the immune cells, which cluster together regardless of the organ of origin). Color code as in Fig. 3a, b. **(b)** Heart lineage tree. As expected, we observe an early separation of myocardial and endocardial lineages. Connections are shown with a dashed line if the exact positions of the corresponding scar clones in the lineage tree are not certain due to limited statistics (see also Supplementary Note 1).

Supplementary Figure 19



Supplementary Figure 19. Zoom into cardiac and pancreatic lineages for one adult. (a) Heart lineage tree. We observe an early separation of myocardial and endocardial lineages. **(b)** Endocrine cells of the primary pancreatic islet. The plot on the left shows the fraction of alpha, beta, delta cells in scar clones 1204, 20 and 35 (normalized per cell type). The plot on the right is a mini tree representation of these three scars. The pies are normalized per scar clone. We observed that delta cells were strongly underrepresented in scar clone 1204, suggesting that a fraction of alpha/beta cells may have a different origin than the majority of pancreatic endocrine cells. The connection to scar 1204 is shown with a dashed line to indicate that the exact position of this scar in the lineage tree is less certain than for the other two scars due to lower cell numbers (see also Supplementary Note 1).

Supplementary Figure 20



Top scar: expected connectivity vs observed connectivity?

Detection rates

$p(41) = 12/(14+23+7+2) \approx 0.26$

$p(86) = 23/(12+14+7+2) \approx 0.66$

Chance of observing scar connections

$p(41\text{-}20) = 1 - (1 - 0.26)^{14} \approx 0.99$

...

$p(41\text{-}33) = 1 - (1 - 0.26)^{2} \approx 0.54$

$p(86\text{-}20) = 1 - (1 - 0.66)^{14} \approx 1.0$

...

$p(86\text{-}33) = 1 - (1 - 0.66)^{2} \approx 0.88$

Chance of measuring at most degree 3

$P(d(41) \leq 3) \approx 0.61$

$P(d(86) \leq 3) \approx 0.12$

Conclusion: scar 41 is the top scar, the left diagram is correct

Supplementary Figure 20. Illustration of tree building corrections as detailed in Supplementary Note 1. (a) Missing connections: in this scar graph, scars 41 and 86 both have degree three. We calculate which of these has the higher probability to be the first created scar. We start with the assumption that scar 41 (left column) or scar 86 (right column) was created first. Based on this assumption we can calculate the detection rate of these scars (in this example all cells have more than one scar, see Supplementary Note 1 for the calculation if this is not the case), and the probabilities of observing the different connections. Using the Poisson binomial distribution, we calculate the probability of observing three or less connections, instead of four. This probability is higher for scar 41, so we conclude that this scar was created first. **(b)** Pruning: In this example, scar 10 was created after scar 85, but we never observe the connection between 10 and 85 (red cross). This may happen at very low cell numbers, i.e. mostly in the last branches of the tree. In this case, we would erroneously conclude that the branches with scars 10 and 85 are siblings, instead of a mother-daughter pair (red arrow). If, additionally, the connection between scar 10 and 78 is also not detected, the branch with scar 10 will even be moved one level higher (purple cross and purple arrow). Such events can be recognized by taking cell numbers into account: If 78 happened before 10, in general there will be more cells with scar 78 than with scar 10. We prune the lineage tree for such branches by removing branches that have less than 12.5% of the average number of cells their siblings have.

Supplementary Table 1: Primer sequences

	Primers for sgRNA synthesis
sgRNA primer 1	TAATACGACTCACTATAGGTGTCCACGTAGTAGTAGCGTTTTAGAGCTAGAAATAGCAAG
sgRNA constant oligo	AAAAGCACCGACTCGGTGCCACTTTTTCAAGTTGATAACGGACTAGCCTTATTTTAACTTGCTATTTCTAGCTCTAAAAC

	Primers for bulk scar detection (forward primers differ only in their barcode sequence)
Reverse primer	GTTCAGAGTTCTACAGTCCGACGATCGTCCTCGTTGTGGGAGGTG
Forward primer 1	CCTTGGCACCCGAGAATTCCACTGGTCGGCGGCACGCTGATCTACAAGG
Forward primer 2	CCTTGGCACCCGAGAATTCCAAGGCTCTACGGCACGCTGATCTACAAGG
Forward primer 3	CCTTGGCACCCGAGAATTCCAGTGGTGCCCGGCACGCTGATCTACAAGG
Forward primer 4	CCTTGGCACCCGAGAATTCCAACGATGTCCGGCACGCTGATCTACAAGG
Forward primer 5	CCTTGGCACCCGAGAATTCCACCGATTGACGGCACGCTGATCTACAAGG
Forward primer 6	CCTTGGCACCCGAGAATTCCAAATTAACCCGGCACGCTGATCTACAAGG
Forward primer 7	CCTTGGCACCCGAGAATTCCACATTACAACGGCACGCTGATCTACAAGG
Forward primer 8	CCTTGGCACCCGAGAATTCCAATTGACCGCGGCACGCTGATCTACAAGG
Forward primer 9	CCTTGGCACCCGAGAATTCCATGTCACGACGGCACGCTGATCTACAAGG
Forward primer 10	CCTTGGCACCCGAGAATTCCAGCTAACTGCGGCACGCTGATCTACAAGG
Forward primer 11	CCTTGGCACCCGAGAATTCCACTTGAGACCGGCACGCTGATCTACAAGG
Forward primer 12	CCTTGGCACCCGAGAATTCCATCTAAGGCCGGCACGCTGATCTACAAGG
Forward primer 13	CCTTGGCACCCGAGAATTCCACGTCATAGCGGCACGCTGATCTACAAGG
Forward primer 14	CCTTGGCACCCGAGAATTCCAACTAATCACGGCACGCTGATCTACAAGG
Forward primer 15	CCTTGGCACCCGAGAATTCCATATTATGGCGGCACGCTGATCTACAAGG
Forward primer 16	CCTTGGCACCCGAGAATTCCATATGCACGCGGCACGCTGATCTACAAGG

	Primers for single cell scar detection (see also the protocol from 10X Genomics Chromium)
Fwd primer 1 targeting RFP sequence	CGGCACGCTGATCTACAAGG
Fwd primer 2 targeting RFP sequence	GTGACTGGAGTTCAGACGTGTGCTCTTCCGATCTGAGTTCAAGACCATCTACATGGCC
Reverse primer	CTACACGACGCTCTTCCGATCT

Supplementary Note 1: Mathematical Supplement

In this document, we will describe the tree building algorithm used in LINNAEUS. Tree building is done in multiple steps: we start by removing possible doublet cells. After that we build the lineage tree of scars, which we prune afterwards. This tree describes the hierarchy of all scarring events that we have measured and can be used to determine the lineage relationship between observed cells. Indeed, as a last step we place all cells in the tree, using their scars to place them on the lowest scarring event possible. We also place all of the cells we flagged as possible doublets that do not have scars in multiple branches (i.e. cells that looked like doublets but actually were not).

1 Doublet recognition

The tree building proper uses a scar graph, in which two scars are connected if they occur together in at least one cell. Cell doublets could lead to spurious connections in the scar graph which will impede the tree building. We therefore seek to remove cell doublets that lead to spurious connections. To do that, we estimate how many times we would see a connection between two scars due to doublets alone, and test the number of times we observe connections between scars against that.

For an experiment with N cells sequenced, of which D doublets, the doublet rate r_d is defined as

$$r_d := \frac{D}{N}. \tag{1}$$

The number of $x - y$ doublets d_{x-y} for N_x cells with scar x and N_y cells with scar y is then

$$d_{x-y} = 2r_d \frac{N_x N_y}{N}, \tag{2}$$

an $x - y$ doublet rate r_{x-y} of

$$r_{x-y} = 2r_d \frac{N_x N_y}{N^2}. \tag{3}$$

For each connection $x - y$, we determine N_{x-y}, the number of cells that express both x and y. We use a binomial test to determine the probability of observing at least that number as doublets, given the doublet rate (3). We only retain scar connections that have a false discovery rate of 0.01. Cells that caused other connections are marked as possible doublets.

2 Tree building

After removal of the spurious scar connections, we can start building the tree. This is done in an iterative manner by determining the first created scar, then determining the scars created after that, and so forth until all scars have been placed.

At the heart of our approach is the observation that the scar that was created first will be present in all cells and will therefore be observed together with all consecutive scars. In terms of the scar graph, the first created scar will have maximal degree (see Fig. 2b). Removing this scar x from the scar graph will create as many connected components as x has daughter scars. Scar x is then placed on top of the scar tree. For each of the resulting connected components, we can again determine and remove the first created scars in these components (we will call the first created scar in a component the *top scar* of that component). These scars are placed in the scar tree as daughters of x. We repeat this procedure until all scars have been placed in the scar tree.

If all scar connections were always observed in all cells, the top scar at every step would be the scar that is connected to all other scars. However, since the experimental scar detection efficiency is smaller than one, we cannot expect to observe every scar connection present. Therefore, we test for every scar x in a connected component whether that scar would have maximum degree if its detection rate was one. More specifically, we test the probability of observing the (experimental) degree of all scars x under the assumption that scar x is the top scar, and select the scar with the highest probability as the true top scar of that connected component. This procedure is sketched in Supplementary Fig. 20a.

To calculate that probability we need to know the detection rate of scar x, assuming that it is the top scar. If x is the top scar, it is present in all cells that have at least one scar in C_x, the set of scars (directly or indirectly) connected to x. Whether we actually observe x in all those cells is determined by its detection rate $p(x)$ that we can calculate as

$$p(x) = p(x|C_x) = \frac{p(x \cap C_x)}{p(C_x)}. \tag{4}$$

Since $p(x \cap C_x) = N_{x \cap C_x}/N_{C_x}$, and $p(C_x) = N_{C_x}/N_{C_x} = 1$, we conclude that

$$p(x) = \frac{N_{x \cap C_x}}{N_{C_x}}. \tag{5}$$

These quantities (the number of cells that express x and a scar connected to x, and the number of cells that express a cell connected to x) can be easily determined from the data, allowing us to calculate the detection rate of any scar x under the assumption that it is the top scar.

We now test the probability of observing the degree of x that we find experimentally, under the null hypothesis of it being maximal. The chance of observing a connection $x - y$ between scars x and y in at least one cell can be calculated as

$$p(x - y) = 1 - p(!(x - y)) = 1 - (1 - p(x))^{N_y} \tag{6}$$

for N_y the number of cells expressing y. The distribution underlying the degree of x is then the *Poisson binomial distribution*, the distribution describing the number of successes in trials with different success probabilities, using $p(x-y)$ for all $y \in C_x$ as success probabilities. We calculate the probability of measuring higher or lower than the observed degree, based on whether the observed degree is higher than the expected value or lower.

The top scar for a component is among the scars for which this probability, the probability of measuring the observed degree when the real degree is maximal, is the highest. We select as top scar x the scar with the highest probability, after filtering out scars that have a detection rate lower than 5%, a value we base on our single-cell experiments with offspring from Cas9-injected fish. Scars y with a higher probability than x but a lower detection rate than 5% are either scars that are lower in the tree, or scars that are on a weakly-expressing integration site. To distinguish between those two cases, we first determine the number of connected components in the scar graph after removing x. If removing y increases the number of connected components, it is likely to be created around the same time as x, but due to its low detection rate we will not be able to place it confidently. We therefore remove such scars from the dataset.

3 Cell placement

Now that we have constructed a scar tree, we can read off the lineage relationship of all cells. We place cells as deep (i.e. as far away from the top of the tree) as possible to maximise the resolution of our lineage tree. This means that for every cell, we determine which of its scars has the highest depth in the tree. The cell is attached to the node corresponding to that scar.

For pie chart representations in the figure, the composition of all nodes, i.e. how many cells of which cell type belong to each node, is calculated cumulatively: we count the cells that are directly connected to that node, and also the cells that are directly connected to the lower nodes.

4 Pruning the tree

Due to dropout events, it is possible that we do not have all information needed to correctly place a scar in the tree. This is especially true for scars that have been created relatively late, and we illustrate this in Supplementary Fig 20b: if we do not observe the connection between scar 10 and scar 85, we will conclude that scars 10 and 85 are siblings. If, furthermore, we do not observe the connection between scars 10 and 78 either, scar 10 would look like a sibling of 78. We can recognise this by calculating the cumulative number of cells for all nodes and comparing these numbers between siblings. If the ratio between the number of cells for a scar node and the number of cells in its biggest sibling becomes less than 1/8, we depict its connection with a dashed line to signify that it is possible this scar actually happened later in the same branch.