Supplementary Material

Computing low-cost rearrangements

For each new occurrence of an architecture, we use simple search or dynamic programming to identify rearrangement solutions with lowest cost. There is a separate procedure for each rearrangement class, as described below. The cost function is the total number of fusion and fission operations. In this section, we consider each architecture to be a sequence or ordered list of domains.

Fusion: Identify parent architectures that are subsequences of the new architecture. Use dynamic programming to select arrangements of parent architectures and new domains, when needed, which can be concatenated into the new architecture using the smallest number of fusion operations. The cost of a solution is equal to the number of sources used minus one.

Fission: Identify parent architectures that are super-sequences of the new architecture. The cost of a solution is one if the new architecture includes an endpoint of the parent architecture. Otherwise, cost is two. Choose the parent architectures with lowest cost.

Insertion: Search for one parent architecture and either a second parent architecture or a new domain, as required by the presence of new domains in the new architecture, such that the second can be inserted into the first. All such pairs are possible solutions with cost three.

Deletion: As in the Fission case, identify parent architectures that are super-sequences of the new architectures. In this case, the pairwise order of domains in the parent architecture should be the same as in the new architecture, but domains are not required to be consecutive in the parent architecture. Use dynamic programming to identify domains that can be removed from the parent architecture with minimum cost to produce the new architecture.

Other: This is the most general case, in which any rearrangement of parent architectures and new domains can be computed. First, we identify all parent architectures that share any domain with the new architecture. Then, we use dynamic programming as in the Fusion case to determine arrangements of domains that can be concatenated to form the new architecture, including all necessary fusion and fission operations into total cost. This procedure differs from Fusion in that for each domain, there may be a cost for splicing the domains from parent architectures. In theory, every domain in the new architecture can be taken from any parent architecture that contains the domain, yielding an exponentially growing number of solutions. To limit the number of solutions, after identifying lowest-cost solutions, we retain those with the smallest number of sources.

Alternative parsimony models

To probe the robustness of our results against changes in the parsimony rule for assigning architectures to internal tree nodes, we compare our MP setting with Dollo parsimony, which is a commonly assumed model, and two variations of MP. In the first variation which we call MP-2child, a parent node with at least two children labeled present for an algorithm will automatically be labeled present, otherwise defaults to standard MP. This variation addresses our concern that MP, typically envisioned with a binary or near-binary tree, is not the ideal rule to assign nodes with higher degree. In the second variation, which we call MP-*X* where *X* has a numeric value, we modify the first pass of the internal node assignment algorithm as follows:

- If architecture is present in more than *max*{*X*%, 50%} of labeled children → label parent *present*.
- If architecture is present in less than *min*{*X*%, 50%} of labeled children → label parent *absent*.
- If architecture is present in between *X*% and 50% of labeled children → label parent *unknown*.

Standard maximum parsimony is equivalent to the case where *X*=50. Decreasing *X* makes it possible to identify an architecture as present in a parent node as present when fewer children contain that architecture. Increasing *X* has the opposite effect, making the rule for propagation of architectures up the tree more strict. Using a range of values rather than exact values circumvents automatic propagation of absent/present status. To illustrate this problem, if we were to use the rule that assigns architecture *A* to a node if more than 49% of children have architecture *A* present, starting with a leaf node containing *A*, every ancestor with only two children will be assigned *A*. This clearly violates the intent of MP to minimize tree cost while making reasonable assignments. We consider values of *X* between 25 and 75 that correspond to specific combinations at child nodes, e.g. *X*=33 refers to the case with 1 child labeled present and 2 children labeled absent.

The immediate result of modifying the parsimony scheme is new assignments of architectures to nodes.

Figure 1 provides a snapshot of the changes induced by various parsimony rules through the number of architectures assigned to selected nodes. Dollo parsimony is expected to yield the greatest changes because its underlying premise is differs from that of MP—it assigns each architecture to be gained at the Last Common Ancestor of all organisms containing that architecture without considering the tree cost, i.e. the number of gains and losses. Dollo parsimony yields five times as many architectures at the root node as MP and multiple times as many architectures at the Archaea, Bacteria, and Eukaryotes nodes. Nearly all architectures are assigned to internal nodes. These confirm that Dollo parsimony tends to label architectures as being more ancient than MP indicates.

The indirect consequence of new architecture assignments at nodes is changes in the solutions we compute for each architecture. While new architectures will appear at different nodes, our main conclusions regarding the distribution of rearrangements over our defined cases does not change over any parsimony method (Figure 2). Even Dollo parsimony produces fairly small changes in the number of architectures at each rearrangement case.

Variants of MP show little difference in node assignment or distribution of rearrangement cases. The one exception is MP-25 and MP-35 assigning more architectures to the root. This outcome is due to these methods allowing an architecture to be assigned to the root if it is present

in only one of Archaea, Bacteria, and Eukaryotes, which in fact is not ideal and suggests that MP-X with X closer to 50 is preferable.

Figure 1: Number of architectures present in each node or group of nodes. To simplify our analysis, we reduce the TOL into Cellular Organisms (root), Archaea, Bacteria, Eukaryotes, all other interior nodes (interim nodes).

Cellular Organisms

Bacteria, Archaea, Eukaryotes

Interim (internal) Nodes

Contemporary Organisms
(11842 architectures)

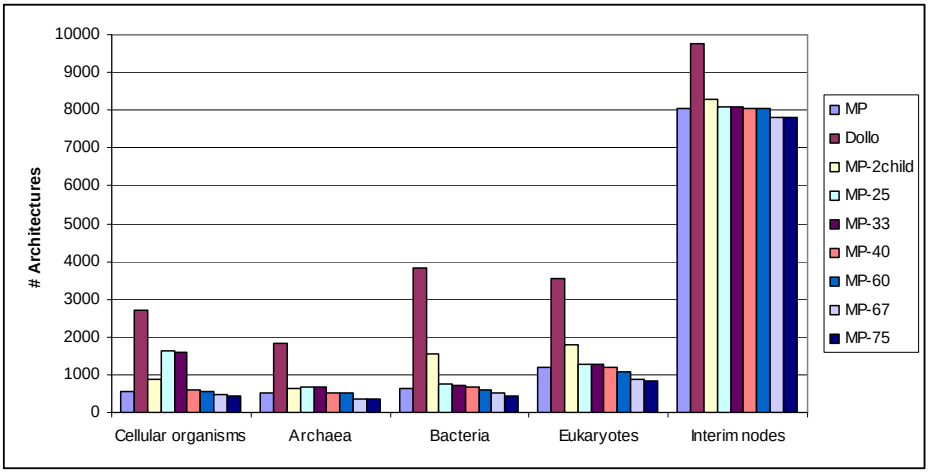
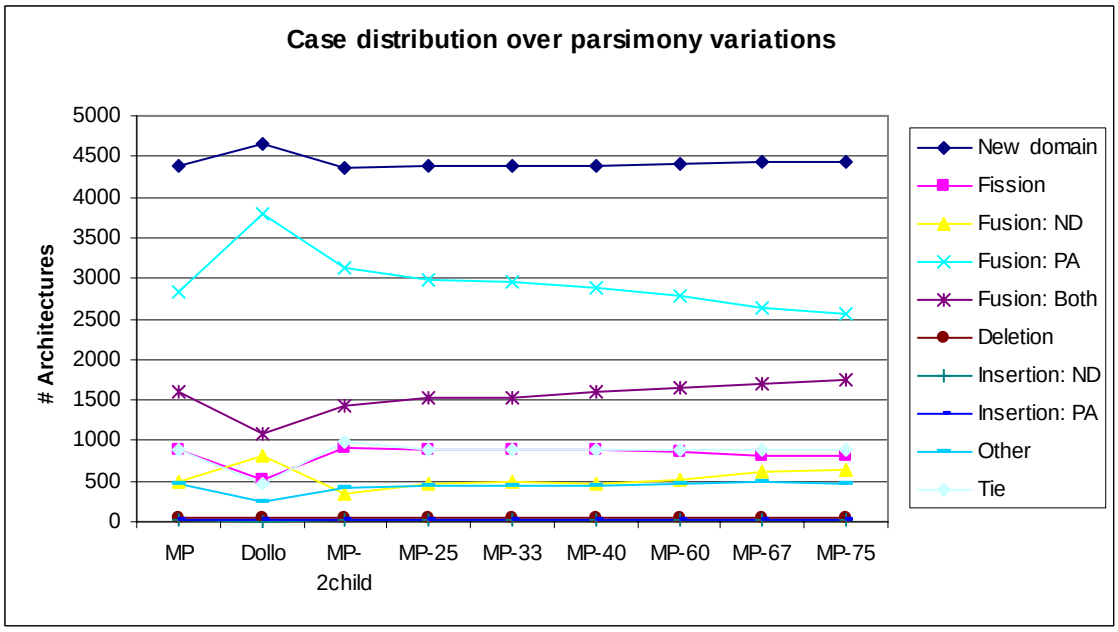


Figure 2: Number of architectures whose most common solution is of each case, with multiple cases among most common solutions counted among the ties. Each line illustrates the values for one rearrangement case.



Supplementary Table 1: Table of species

Anopheles gambiae	Helicobacter hepaticus ATCC 51449
Arabidopsis thaliana	Idiomarina loihiensis L2TR
Caenorhabditis elegans	Lactobacillus plantarum WCFS1
Drosophila melanogaster	Lactococcus lactis subsp. lactis Il1403
Encephalitozoon cuniculi	Legionella pneumophila subsp. pneumophila str.
Saccharomyces cerevisiae	Philadelphia 1
Plasmodium falciparum	Leifsonia xyli subsp. xyli str. CTCB07
Schizosaccharomyces pombe	Leptospira interrogans serovar Copenhageni str.
Homo sapiens	Fiocruz L1-130
Mus musculus	Listeria monocytogenes str. 4b F2365
Rattus norvegicus	Mannheimia succiniciproducens MBEL55E
Danio rerio	Mesoplasma florum L1
Oryza sativa	Mesorhizobium loti MAFF303099
Filobasidiella neoformans	Methanocaldococcus jannaschii DSM 2661
Entamoeba histolytica	Methanococcus maripaludis S2
Eremothecium gossypii	Methanopyrus kandleri AV19
Kluyveromyces lactis	Methanosarcina acetivorans C2A
Leishmania major	Methanothermobacter thermautotrophicus str. Delta H
Trypanosoma cruzi	Methylococcus capsulatus str. Bath
Yarrowia lipolytica	Mycobacterium avium subsp. paratuberculosis K-10
Plasmodium berghei	Mycoplasma penetrans HF-2
Plasmodium chabaudi	Nanoarchaeum equitans Kin4-M
Trypanosoma brucei	Natronomonas pharaonis DSM 2160
Caenorhabditis briggsae	Neisseria meningitidis Z2491
Bos taurus	Nitrobacter winogradskyi Nb-255
Canis familiaris	Nitrosococcus oceani ATCC 19707
Dictyostelium discoideum	Nitrosomonas europaea ATCC 19718
Apis mellifera	Nocardia farcinica IFM 10152
Strongylocentrotus purpuratus	Nostoc sp. PCC 7120
Gallus gallus	Oceanobacillus iheyensis HTE831
Xenopus laevis	Onion yellows phytoplasma OY-M
Acinetobacter sp. ADP1	Parachlamydia sp. UWE25
Aeropyrum pernix K1	Pasteurella multocida subsp. multocida str. Pm70
Agrobacterium tumefaciens str. C58	Pelobacter carbinolicus DSM 2380
Anabaena variabilis ATCC 29413	Photobacterium profundum SS9
Anaplasma marginale str. St. Maries	Photorhabdus luminescens subsp. laumondii TTO1
Aquifex aeolicus VF5	Picrophilus torridus DSM 9790
Archaeoglobus fulgidus DSM 4304	Porphyromonas gingivalis W83
Azoarcus sp. EbN1	Prochlorococcus marinus str. MIT 9313
Bacillus cereus ATCC 14579	Propionibacterium acnes KPA171202
Bacteroides fragilis NCTC 9343	Pseudoalteromonas haloplanktis TAC125
Bartonella henselae str. Houston-1	Pseudomonas fluorescens Pf-5
Bdellovibrio bacteriovorus HD100	Psychrobacter arcticus 273-4
Bifidobacterium longum NCC2705	Pyrobaculum aerophilum str. IM2
Bordetella bronchiseptica RB50	Pyrococcus furiosus DSM 3638
Borrelia garinii PBi	Ralstonia eutropha JMP134
Bradyrhizobium japonicum USDA 110	Rhodobacter sphaeroides 2.4.1
Brucella suis 1330	Rhodopirellula baltica SH 1
Buchnera aphidicola str. APS (Acyrthosiphon pisum)	Rhodopseudomonas palustris CGA009
	Rickettsia felis URRWXCal2

Burkholderia sp. 383	Salmonella typhimurium LT2
Campylobacter jejuni RM1221	Shewanella oneidensis MR-1
Candidatus Pelagibacter ubique HTCC1062	Shigella flexneri 2a str. 301
Carboxydothermus hydrogenoformans Z-2901	Silicibacter pomeroyi DSS-3
Caulobacter crescentus CB15	Sinorhizobium meliloti 1021
Chlamydia trachomatis D/UW-3/CX	Staphylococcus aureus subsp. aureus MSSA476
Chlamydophila pneumoniae TW-183	Streptococcus pneumoniae TIGR4
Chlorobium tepidum TLS	Streptomyces coelicolor A3(2)
Chromobacterium violaceum ATCC 12472	Sulfolobus tokodaii str. 7
Clostridium acetobutylicum ATCC 824	Symbiobacterium thermophilum IAM 14863
Colwellia psychrerythraea 34H	Synechococcus elongatus PCC 6301
Corynebacterium glutamicum ATCC 13032	Synechocystis sp. PCC 6803
Coxiella burnetii RSA 493	Thermoanaerobacter tengcongensis MB4
Dechloromonas aromatica RCB	Thermobifida fusca YX
Dehalococcoides ethenogenes 195	Thermococcus kodakarensis KOD1
Deinococcus radiodurans R1	Thermoplasma volcanium GSS1
Desulfotalea psychrophila LSv54	Thermosynechococcus elongatus BP-1
Desulfovibrio vulgaris subsp. vulgaris str. Hildenborough	Thermotoga maritima MSB8
	Thermus thermophilus HB8
Ehrlichia ruminantium str. Welgevonden	Thiobacillus denitrificans ATCC 25259
Enterococcus faecalis V583	Thiomicrospira denitrificans ATCC 33889
Erwinia carotovora subsp. atroseptica SCRI1043	Treponema denticola ATCC 35405
Escherichia coli CFT073	Tropheryma whipplei TW08/27
Francisella tularensis subsp. tularensis SCHU S4	Ureaplasma parvum serovar 3 str. ATCC 700970
	Vibrio vulnificus YJ016
Fusobacterium nucleatum subsp. nucleatum ATCC 25586	Wigglesworthia glossinidia endosymbiont of Glossina brevipalpis
Geobacillus kaustophilus HTA426	Wolbachia endosymbiont of Drosophila melanogaster
Geobacter metallireducens GS-15	Xanthomonas campestris pv. vesicatoria str. 85-10
Gloeobacter violaceus PCC 7421	Xylella fastidiosa 9a5c
Gluconobacter oxydans 621H	Yersinia pestis biovar Medievalis str. 91001
Haemophilus influenzae 86-028NP	Zymomonas mobilis subsp. mobilis ZM4
Haloarcula marismortui ATCC 43049	
Halobacterium sp. NRC-1	