

Supplemental Material:

Resource heterogeneity structures aquatic bacterial communities

Mario E. Muscarella, Claudia M. Boot, Corey D. Broeckling, Jay T. Lennon

DOM Annotations — The following parameters were used when annotating DOM components using RAMClustR (Broeckling *et al.*, 2014):

- 1) Parameters for the *do.findmain* function were set to: mode = negative, mzabs.error = 0.01, ppm.error = 10, ads = [M-H]- [M+Na-2H]- [M+K-2H]- [M+CH₂O₂-H]- [2M-H]- [2M+Na-2H]- [2M+K-2H]- [2M+CH₂O₂- H]- [3M-H]- [3M+Na-2H]- [3M+K-2H]- [3M+CH₂O₂- H]-, nls = [M-H-NH₃]- [M-H-H₂O]- [M-H-COCH₂]- [M-H-CO₂]- [M-H-NH₃-CO₂]- [M-H-NH₃-HCOOH]- [M-H-NH₃-H₂O]- [M-H-NH₃-COCH₂]- [M-H-S]- [M-H-S-NH₃-HCOOH]- [M-H-H₄O₂]- [M-H-CH₂]- [M-H-O]- [M-H-C₂H₂]- [M-H-C₂H₄]- [M-H-CO]- [M-H-C₃H₆]- [M-H-C₂H₄O]- [M-H-C₄H₆]- [M-H-C₃H₄O]- [M-H-C₄H₈]- [M-H-C₅H₈]- [M-H-C₄H₆O]- [M-H-C₅H₁₀]- [M-H-C₆H₁₂]- [M-H-C₄H₈O₂]- [M-H-H₂O-HCOOH]- [M-H-CH₄]- [M-H-CH₂O]- [M-H-C₂H₆]- [M-H-CH₃OH]- [M-H-C₃H₄]- [M-H-C₃H₆O]- [M-H-CO₂-C₃H₆]- [M-H-SO₃]- [M-H-SO₃-H₂O]- [M-H-SO₃-H₂O-NH₃]- [M-H-NH₃-C₃H₄]- [M-H-H₂O-CO₂]- [M-H-H₂O-H₂O-C₂H₄O]- [M-H-NH₃-CO-CO]- [M-H-NH₃-CO-COCH₂]- [M-H-C₈H₆O]- [M-H-C₈H₆O-NH₃]- [M-H-C₈H₆O-H₂O]- [M-H-C₂H₂O₂]- [M-H-C₂H₄O₂]- [M-H-C₅H₈O]- [M-H-NH₃-CO₂-CH₂O]- [M-H-NH₃-CO₂-NH₃-H₂O]- [M-H-NH₃-CO₂-C₃H₄O]- [M-H-NH₃-CO₂-C₅H₈]- [M-H-HCOOH-HCOOH]- [M-H-C₂H₄-CO₂]- [M-H-C₂H₄-HCOOH]- [M-H-NH₃-H₂O-H₂O]- [M-H-H₂O-C₂H₂O₂]- [M-H-COCH₂-C₄H₈]- [M-H-NH₃-NH₃-C₃H₄]- [M-H-C₂H₄O₂-CH₃OH]- [M-H-C₃H₆O-CH₃OH]- [M-H-NH₃-CO-COCH₂-C₄H₆O]- [M-H-C₄H₆-H₂O]- [M-H-C₄H₆-C₂H₄]- [M-H-C₄H₆-NH₃-H₂O]- [M-H-C₄H₆-COCH₂]- [M-H-C₄H₆-C₄H₆O]- [M-H-C₃H₄O-C₄H₆]- [M-H-C₃H₄O-C₄H₈O₂]-

[M-H-C4H8-C4H6]- [M-H-NH3-HCOOH-CH3OH]- [M-H-NH3-C2H6]- [M-H-NH3-C8H6O-CH2]- [M-H-NH3-C3H4-COCH2]- [M-H-C3H9N]- [M-H-C3H9N-C2H4O2]- [M-H-C6H10O7]- [M-H-C6H10O7+H2O]- [M-H-C6H10O7-(H2O)2]- [M-H-C6H12O6]- [M-H-C6H12O6+H2O]- [M-H-C6H12O6-H2O]- [M-H-C5H10O5]- [M-H-C5H10O5+H2O]- [M-H-C5H10O5-H2O]- [M-H-C2H8NO4P]- [M-H-(H2O)3-CO]- [M-H-C6H13NO2]- [M-H-C5H11NO2]- [M-H-CH3S]- [M-H-C8H8O2]- [M-H-C12H22O11]- [M-H-C12H24O12]- [M-H-C12H20O10]-, scoring = auto, and use.z = TRUE

- 2) MSFinder formula results were imported with database priority set to: UNPD MINE
ChEBI T3DB STOFF HMDB KNApSAcK FooDB NANPDB DrugBank SMPDB
YMDB BMDB ECMDB PubChem PlantCyc Urine Serum LipidMAPS Feces NA CSF
Saliva
- 3) RAMClustR *annotate* function annotation priority was assigned from highest priority to lowest: MSFinder structure, MSFinder formula, interpretMSSpectrum
- 4) Synonyms were retrieved using the RAMClustR *get.synonyms* function within RAMClustR, calling the Chemical Translation Service (Wohlgemuth *et al.*, 2010) as well as database API for Pubchem (Kim *et al.*, 2019), LipidMaps (Fahy *et al.*, 2007), and HMDB (Wishart *et al.*, 2018), when appropriate

Sequencing Results — After quality controls and removing singletons, we had recovered the following numbers of sequences from each site based on the active community:

Ann Lake - 96963

Canyon Lake - 79973

Howe Lake - 64182

Ives Lake - 101741

Lily Lake - 150702

Mountain Lake - 96933

Pony Lake - 127352

Rush Lake - 84750

Second Pine Lake - 89873

Upper Pine Lake - 111968

Transformed Data Table — We Box-Cox transformed resource concentration and OTU richness to meet the model assumptions regarding the distribution of residuals. We used a Shapiro Test (*shapiro.test* function in the stats R package) to determine while model parameters were significantly different from normal and therefore needed to be transformed.

DOC: $W = 0.63753$, $p\text{-value} = 0.0001555$

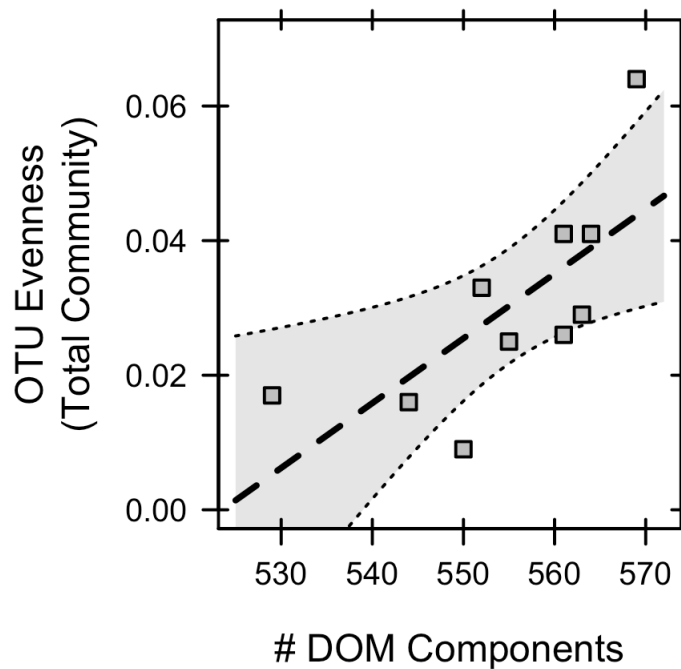
OTU Richness: $W = 0.69123$, $p\text{-value} = 0.0006898$

Transformed values have been Box-Cox transformed and also centered and scaled

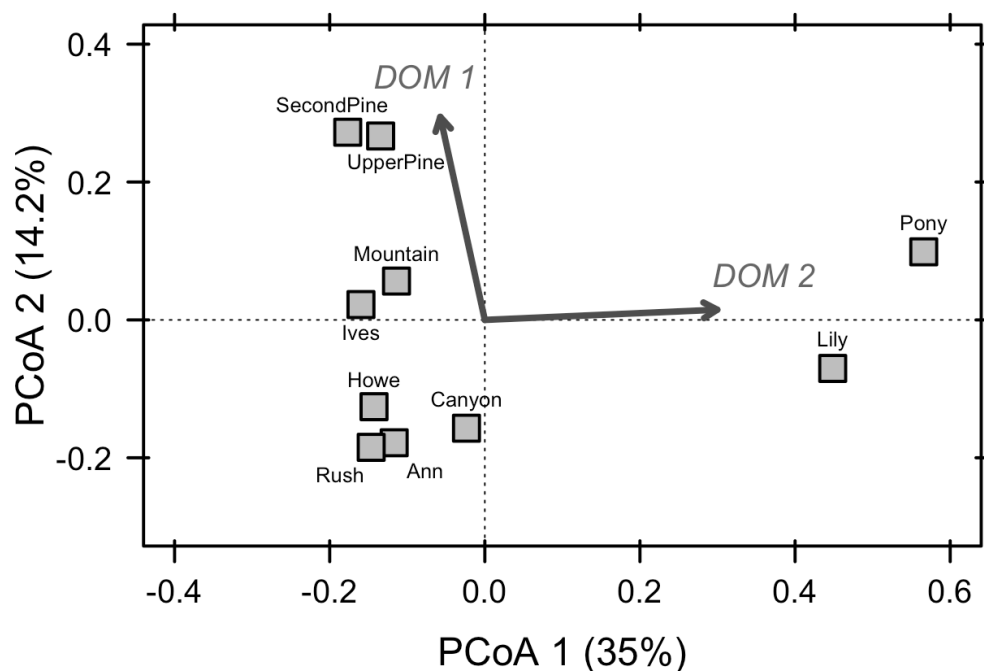
Lake	DOC (mg C L ⁻¹)	Transformed DOC	Rarified OTU Richness	Transformed OTU Richness
Ann	5.97	-0.44	933	0.32
Canyon	7.23	0.08	1046	0.69
Howe	7.04	0.02	808	-0.34
Ives	6.91	-0.02	766	-0.65
Lily	14.35	1.25	1710	1.47
Mountain	5.27	-0.86	830	-0.20
Pony	28.99	1.77	2344	1.64
Rush	4.22	-1.77	706	-1.22
Second Pine	6.26	-0.30	743	-0.85
Upper Pine	7.84	0.28	739	-0.88

Total Bacterial Community — In addition to sequencing the active bacterial community, we sequenced the total community. We used a DNA based approach to characterize the composition of the total by sequencing the 16S rRNA gene. We extracted total nucleic acids using the MoBio Power Water RNA extraction kit (Carlsbad, CA). Nucleic acid extracts were cleaned via ethanol precipitation and DNA extracts were treated with RNase (Invitrogen) to degrade residual RNA. Once DNA samples were cleaned and quantified, we amplified, sequenced, and processed the data in the exact same pipeline as the RNA based approach. In addition, both the total and active community samples were clustered into OTUs at the same time so that OTUs could be compared. All initial sequence processing was completed using the software package mothur (version 1.40.5; Schloss *et al.*, 2009).

Total Community and Resource-Diversity Relationship — First, we tested the hypothesis that resource heterogeneity affects bacterial community evenness to compare with the results from the active community. We used linear models to determine if the number of DOM components would affect the evenness of bacterial communities. We removed singleton OTUs (i.e., those found only once across all samples), and we subsampled using rarefaction to correct for differences in sample size due to sequencing depth (Hughes and Hellmann, 2005; James and Rathbun, 1981). We calculated species evenness using Simpson’s evenness (Smith and Wilson, 1996). We calculated the number of DOM components as the number of distinct DOM peaks observed in each sample. As predicted and observed based on the active microbial community, bacterial evenness was affected by the number of DOM components. OTU evenness was positively related to the number of DOM components ($r^2 = 0.58$, $p = 0.01$).



Next, we tested the hypothesis that DOM heterogeneity affects community beta-diversity by comparing DOM composition to total bacterial community composition. We used distance-based redundancy analysis (dbRDA; Legendre and Anderson, 1999) to test for relationships between resource composition and community composition, and we used principal coordinates analysis (PCoA), based on relative abundances and Bray-Curtis dissimilarity, to decompose DOM composition into orthogonal linear components (Legendre and Legendre, 2012). To represent the DOM composition, we used the DOM PCoA axis scores for each sample. As the response in the dbRDA model, we relativized OTU abundances and used Bray-Curtis distances to compare community composition across samples. Significance tests of our dbRDA model were conducted based on 10,000 permutations. All calculations were done in the R statistical environment (R Core Team, 2012) using the ‘vegan’ package (Oksanen *et al.*, 2013). As predicted and observed based on the active microbial community, we observed relationships between DOM composition and the beta-diversity of the total community. Based on the dbRDA models, DOM composition explained 45 % of the variation in bacterial community composition ($p = 0.03$).



Supplemental TABLES

Table S1. DOM Component Annotations. Annotations are provided for the 172 influential DOM components. DIM1 and DIM2 refer to the contribution of the DOM component to the DOM Principal Coordinates Analysis (Fig. 1; Fig. S3). Each component includes its retention time, an inferred molecular mass and formula, full elemental stoichiometry (number of C, H, N, O, P, and S atoms per molecule) and stoichiometric ratio (C:N:P). In addition, we provide some molecular summary information including, H/C which indicates the amount of hydrogen saturation, O/C which describes the degree of oxygenation, N/C which describes the nitrogen content (Roth *et al.*, 2015). In addition, we provide the CHO index which is a proxy for the degradation potential (values between -1 and 0 are thought to be highly degradable) (Mann *et al.*, 2015), the nominal oxidation state of carbon (NOSC) which provides information on the biogeochemical reactivity (Riedel *et al.*, 2012), double bond equivalents (DBE), the difference between DBE and number of oxygen atoms (DBE-O) which is a rough measure for C = C bonds because it omits any possible C = O bonds (D'Andrilli *et al.*, 2010). In addition, components molecular annotations are provided and organized based on chemical ontology and a PubChem URL is provided if available.

Table S2. The complete list of cosmopolitan taxa including taxonomy. Each OTU is classified based on the RDP taxonomy down to genus. OTUs are listed in order of total abundance across all sites.

Table S3. The complete list of resource generalist taxa including taxonomy. Each OTU is classified based on the RDP taxonomy down to genus. OTUs are listed in order of total abundance across all sites.

Supplemental Figures

Figure S1. Study system map of the Huron Mountain area. Image retrieved from Google Maps.

Lakes used in this study are labeled.

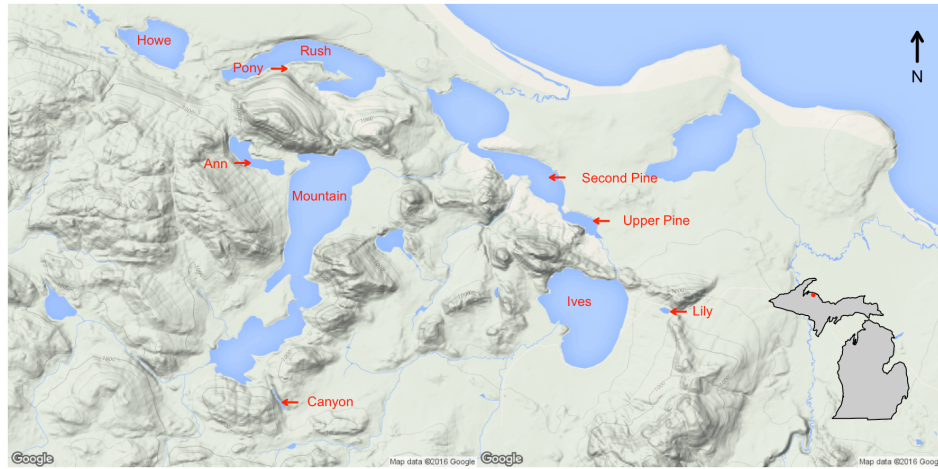


Figure S2. Resource Correlations. The concentrations of dissolved organic carbon (DOC) and total nitrogen (TN) are highly correlated across lakes ($\rho = 0.97, p < 0.001$).

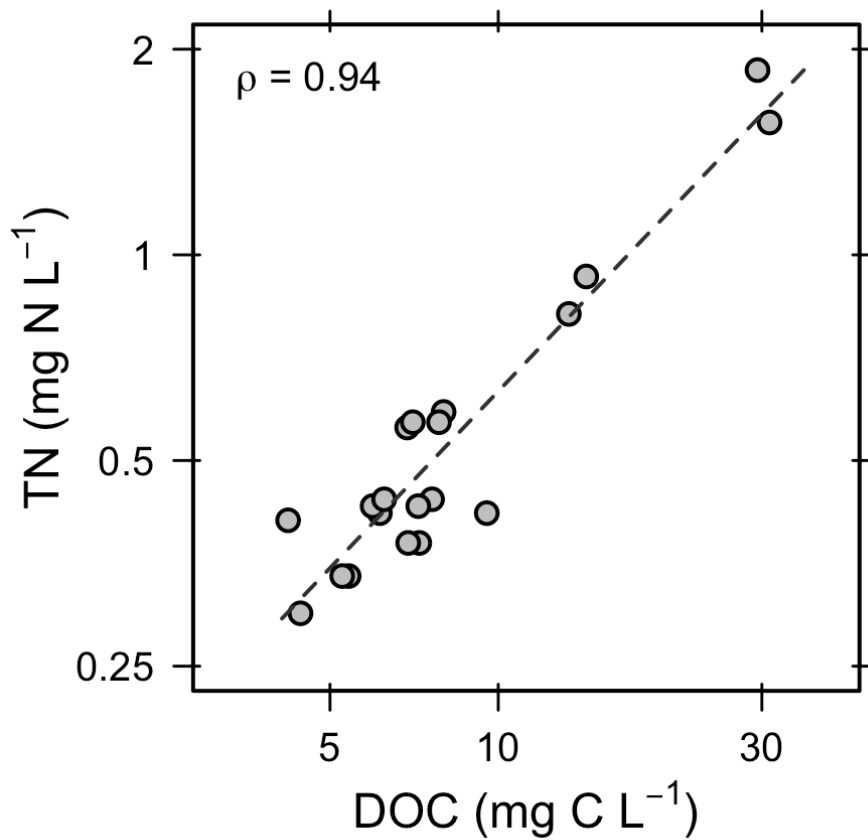


Figure S3. Principal coordinates analysis (PCoA) ordination of dissolved organic matter (DOM) with influential DOM components. We identified influential DOM components as those correlated ($\rho > 0.70$) with variation in the DOM PCoA axes. The distances between symbols represent the dissimilarity in DOM composition between lakes. Symbol sizes reflect variation in the concentration of dissolved organic carbon. Red vectors represent the correlation between each influential DOM component and the PCoA axes.

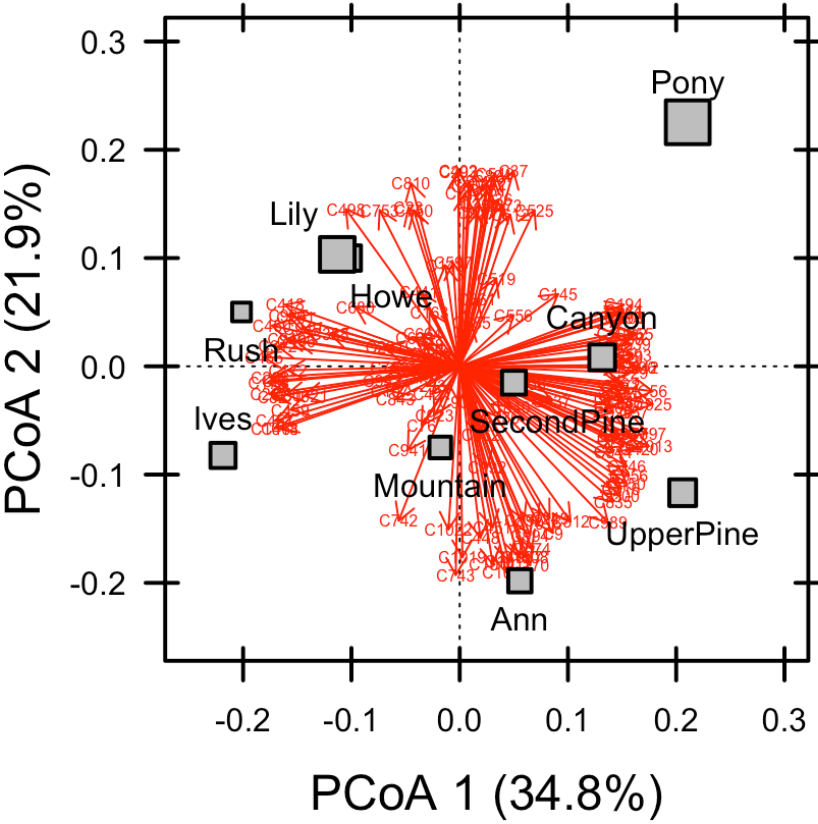


Figure S4. Bacterial community diversity relationships with resource concentration and the number of DOM components using untransformed data.

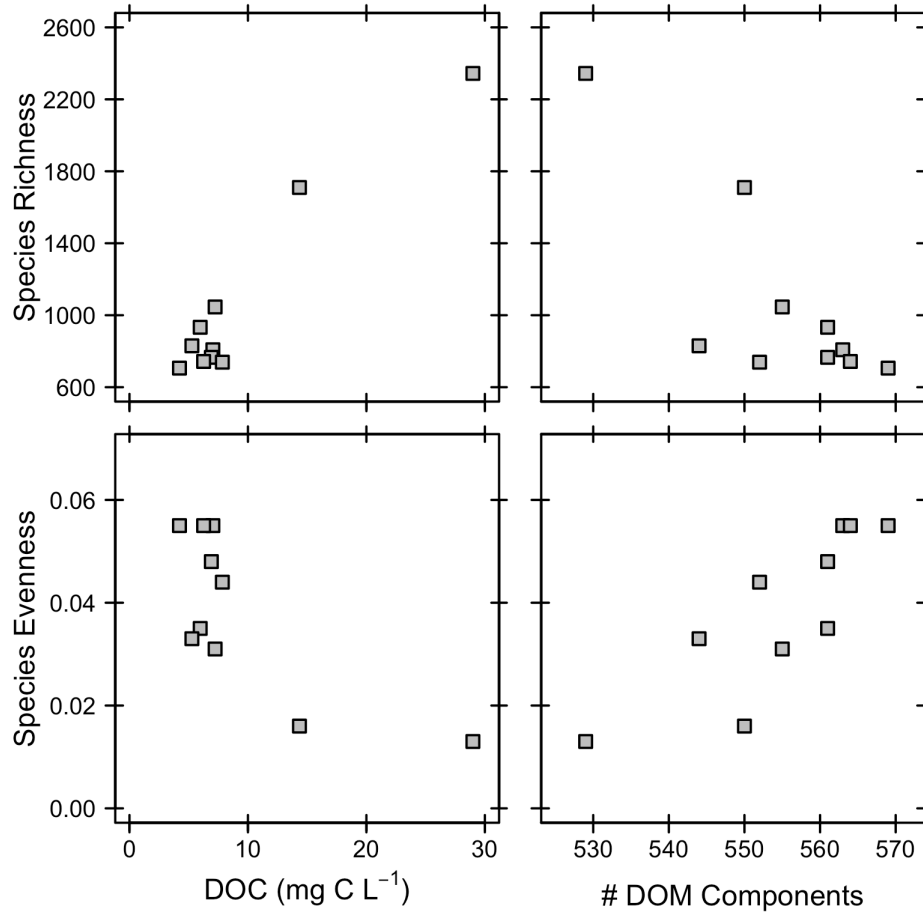
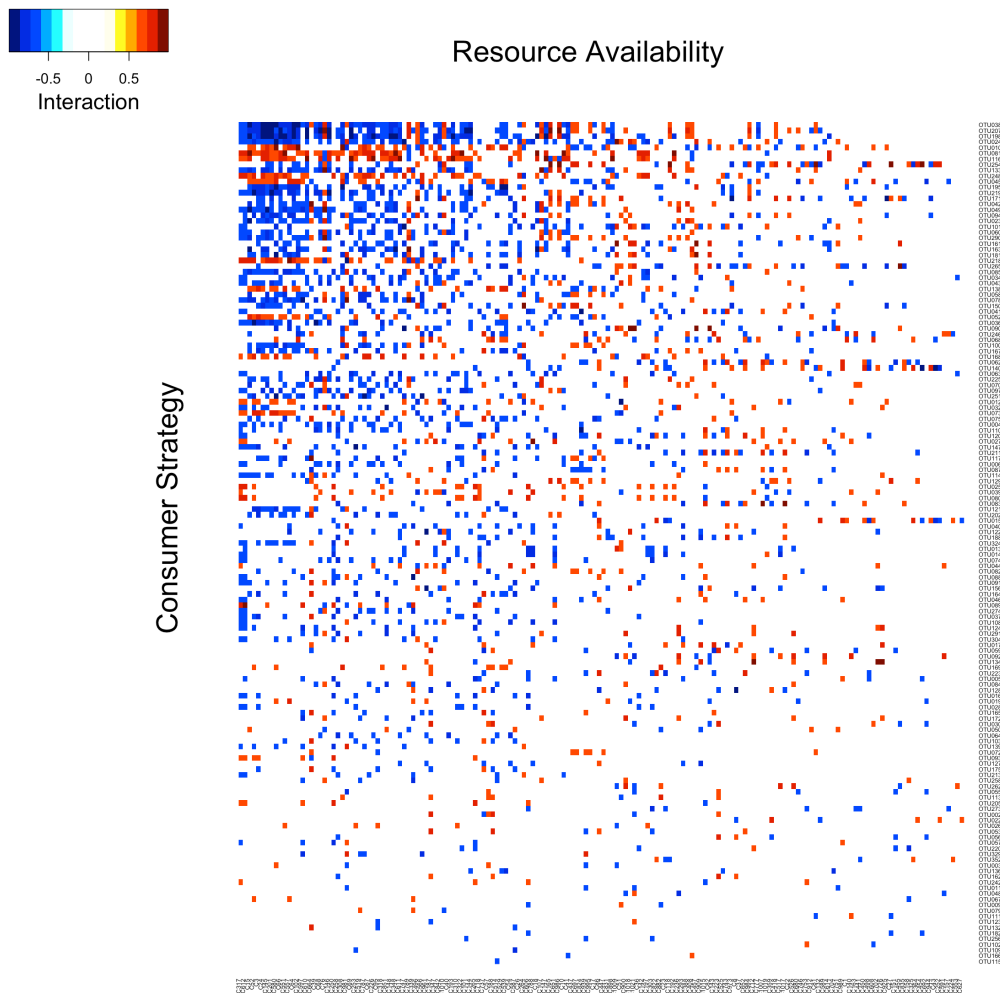


Figure S5. Bacteria-DOM Interactions. Interactions were calculated based on Spearman's Rank-Order Regressions between bacterial OTU and DOM component abundances across sites. Significant interactions were determined using a randomization approach, and only significant interactions (i.e., those with $p < 0.05$ based on 1000 randomizations) with coefficients of determination $> |0.7|$ were considered. Interactions are ordered across columns and rows based on the number of interactions present. Therefore, rows are ordered from generalist (most interactions; on right) to specialist (least interactions; on left) – Consumer Strategy. Likewise, columns are ordered from available (most interactions; on top) to restrictive (least interactions; on bottom) – Resource Availability.



References

- Broeckling CD, Afsar FA, Neumann S, Prenni JE. (2014). RAMClust: a Novel feature clustering method enables spectral-matching-based annotation for metabolomics data. *Anal Chem* **86**: 6812–6817.
- D’Andrilli J, Chanton JP, Glaser PH, Cooper WT. (2010). Characterization of dissolved organic matter in northern peatland soil porewaters by ultra high resolution mass spectrometry. *Org Geochem* **41**: 791–799.
- Fahy E, Sud M, Cotter D, Subramaniam S. (2007). LIPID MAPS online tools for lipid research. *Nucleic Acids Res* **35**: 606–612.
- Hughes JB, Hellmann JJ. (2005). The application of rarefaction techniques to molecular inventories of microbial diversity. *Methods Enzymol* **397**: 292–308.
- James FC, Rathbun S. (1981). Rarefaction, relative abundance, and diversity of avian communities. *Auk* **98**: 785–800.
- Kim S, Chen J, Cheng T, Gindulyte A, He J, He S, *et al.* (2019). PubChem 2019 update: improved access to chemical data. *Nucleic Acids Res* **47**: D1102–D1109.
- Legendre P, Anderson MJ. (1999). Distance-based redundancy analysis: testing multispecies responses in multifactorial ecological experiments. *Ecol Monogr* **69**: 1–24.
- Legendre P, Legendre LFJ. (2012). *Numerical Ecology*. Elsevier.
- Mann BF, Chen H, Herndon EM, Chu RK, Tolic N, Portier EF, *et al.* (2015). Indexing permafrost soil organic matter degradation using high-resolution mass spectrometry. *PLoS One* **10**: 1–16.
- Oksanen J, Blanchet FG, Kindt R, Legendre P, Minchin PR, O’Hara RB, *et al.* (2013). *Vegan*: community ecology package. <http://cran.r-project.org/package=vegan>.
- R Core Team. (2012). *R: A language and environment for statistical computing*. <http://www.r-project.org/>.

Riedel T, Biester H, Dittmar T. (2012). Molecular fractionation of dissolved organic matter with metal salts. *Environ Sci Technol* **46**: 4419–4426.

Roth VN, Dittmar T, Gaupp R, Gleixner G. (2015). The molecular composition of dissolved organic matter in forest soils as a function of pH and temperature. *PLoS One* **10**: 1–23.

Schloss PD, Westcott SL, Ryabin T, Hall JR, Hartmann M, Hollister EB, *et al.* (2009). Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities. *Appl Environ Microbiol* **75**: 7537–7541.

Smith B, Wilson JB. (1996). A consumer's guide to evenness indices. *Oikos* **76**: 70–82.

Wishart DS, Feunang YD, Marcu A, Guo AC, Liang K, Vázquez-Fresno R, *et al.* (2018). HMDB 4.0: The human metabolome database for 2018. *Nucleic Acids Res* **46**: D608–D617.

Wohlgemuth G, Haldiya PK, Willighagen E, Kind T, Fiehn O. (2010). The chemical translation service—a web-based tool to improve standardization of metabolomic reports. *Bioinformatics* **26**: 2647–2648.

Mothur Pipeline

```
make.contigs(file=HMWF.files, insert=30, deltaq=6, processors=8)
summary.seqs(fasta=HMWF.trim.contigs.fasta, processors=8)
screen.seqs(fasta=HMWF.trim.contigs.fasta, group=HMWF.contigs.groups,
summary=HMWF.trim.contigs.summary, maxambig=0, maxlength=275, processors=8)
remove.seqs(accnos=HMWF.trim.contigs.bad.accnos, group=HMWF.contigs.groups)
system(mv HMWF.contigs.pick.groups HMWF.contigs.good.groups)
unique.seqs(fasta=HMWF.trim.contigs.good.fasta)
summary.seqs(fasta=HMWF.trim.contigs.good.unique.fasta, processors=8)
count.seqs(name=HMWF.trim.contigs.good.names, group=HMWF.contigs.good.groups)
summary.seqs(fasta=HMWF.trim.contigs.good.unique.fasta,
count=HMWF.trim.contigs.good.count_table, processors=8)
align.seqs(fasta=HMWF.trim.contigs.good.unique.fasta, reference=silva.v4.fasta, flip=F,
processors=8)
summary.seqs(fasta=HMWF.trim.contigs.good.unique.align, processors=8)
summary.seqs(fasta=HMWF.trim.contigs.good.unique.flip.accnos, processors=8)
screen.seqs(fasta=HMWF.trim.contigs.good.unique.align,
count=HMWF.trim.contigs.good.count_table, minlength=250, maxhomop=8,
processors=8)
remove.seqs(accnos=HMWF.trim.contigs.good.unique.bad.accnos,
count=HMWF.trim.contigs.good.count_table)
system(mv HMWF.trim.contigs.good.pick.count_table
HMWF.trim.contigs.good.good.count_table)
count.groups(count=HMWF.trim.contigs.good.good.count_table)
```



```

summary.seqs(fasta=HMWF.trim.contigs.good.unique.good.align,
count=HMWF.trim.contigs.good.good.count_table, processors=8)
screen.seqs(fasta=HMWF.trim.contigs.good.unique.good.align,
count=HMWF.trim.contigs.good.good.count_table, maxambig=0, maxhomop=8,
minlength=250, start=1965, end=11553, processors=8)
remove.seqs(accnos=HMWF.trim.contigs.good.unique.good.bad.accnos,
count=HMWF.trim.contigs.good.good.count_table)
system(mv HMWF.trim.contigs.good.good.pick.count_table
HMWF.trim.contigs.good.good.good.count_table)
summary.seqs(fasta=HMWF.trim.contigs.good.unique.good.good.align,
count=HMWF.trim.contigs.good.good.good.count_table, processors=8)
filter.seqs(fasta=HMWF.trim.contigs.good.unique.good.good.align, vertical=T, trump=.)
summary.seqs(fasta=HMWF.trim.contigs.good.unique.good.good.filter.fasta,count=HM
WF.trim.contigs.good.good.good.count_table, processors=8)
unique.seqs(fasta=HMWF.trim.contigs.good.unique.good.good.filter.fasta,
count=HMWF.trim.contigs.good.good.good.count_table)
summary.seqs(fasta=HMWF.trim.contigs.good.unique.good.good.filter.unique.fasta,
count=HMWF.trim.contigs.good.unique.good.good.filter.count_table, processors=8)
pre.cluster(fasta=HMWF.trim.contigs.good.unique.good.good.filter.unique.fasta,
count=HMWF.trim.contigs.good.unique.good.good.filter.count_table, diffs=2,
processors=8)
summary.seqs(fasta=HMWF.trim.contigs.good.unique.good.good.filter.unique.precluster
.fasta,
count=HMWF.trim.contigs.good.unique.good.good.filter.unique.precluster.count_table,
processors=8)

```

```
chimera.vsearch(fasta=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.fasta,
count=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.count_table,
dereplicate=t, processors=8)
remove.seqs(fasta=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.fasta,
accnos=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.denovo.vsearch.accnos)
summary.seqs(fasta=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.pick.fasta,
count=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.denovo.vsearch.pick.count_table, processors=8)
summary.seqs(fasta=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.pick.fasta,
count=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.denovo.vsearch.pick.count_table, processors=8)
classify.seqs(fasta=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.pick.fasta,
count=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.denovo.vsearch.pick.count_table, reference=trainset16_022016.pds.fasta,
taxonomy=trainset16_022016.pds.tax, cutoff=80, probs=f, processors=8)
remove.lineage(fasta=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.pick.fasta,
count=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.denovo.vsearch.pick.count_table,
```

```
taxonomy=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.pick.pds.
wang.taxonomy, taxon=Archaea-Mitochondria-Cyanobacteria/Chloroplast;Chloroplast-
unknown-Eukaryota)
summary.seqs(fasta=HMWF.trim.contigs.good.unique.good.filter.unique.precluster
.pick.pick.fasta,
count=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.denovo.vsear
ch.pick.pick.count_table, processors=8)
classify.seqs(fasta=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.p
ick.pick.fasta,
count=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.denovo.vsear
ch.pick.pick.count_table, reference=trainset16_022016.pds.fasta,
taxonomy=trainset16_022016.pds.tax, cutoff=95, probs=t, processors=8)
remove.lineage(fasta=HMWF.trim.contigs.good.unique.good.filter.unique.precluste
r.pick.pick.fasta,
count=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.denovo.vsear
ch.pick.pick.count_table,
taxonomy=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.
pds.wang.taxonomy, taxon=unknown)
summary.seqs(fasta=HMWF.trim.contigs.good.unique.good.filter.unique.precluster
.pick.pick.pick.fasta,
count=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.denovo.vsear
ch.pick.pick.pick.count_table, processors=8)
classify.seqs(fasta=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.p
ick.pick.pick.fasta,
count=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.denovo.vsear
```

```
ch.pick.pick.pick.count_table, reference=trainset16_022016.pds.fasta,
taxonomy=trainset16_022016.pds.tax, cutoff=80, probs=t, processors=8)
cluster.split(fasta=HMWF.trim.contigs.good.unique.good.good.filter.unique.precluster.p
ick.pick.pick.fasta,
taxonomy=HMWF.trim.contigs.good.unique.good.good.filter.unique.precluster.pick.pick.
pick.pds.wang.taxonomy,
count=HMWF.trim.contigs.good.unique.good.good.filter.unique.precluster.denovo.vsear
ch.pick.pick.pick.count_table, splitmethod=classify, taxlevel=4, method=opti,
processors=8)
make.shared(list=HMWF.trim.contigs.good.unique.good.good.filter.unique.precluster.pic
k.pick.pick.opti_mcc.list,
count=HMWF.trim.contigs.good.unique.good.good.filter.unique.precluster.denovo.vsear
ch.pick.pick.pick.count_table, label=0.03)
classify.seqs(fasta=HMWF.trim.contigs.good.unique.good.good.filter.unique.precluster.p
ick.pick.pick.fasta,
count=HMWF.trim.contigs.good.unique.good.good.filter.unique.precluster.denovo.vsear
ch.pick.pick.pick.count_table, reference=trainset16_022016.pds.fasta,
taxonomy=trainset16_022016.pds.tax, cutoff=25, probs=t, processors=8)
classify.otu(list=HMWF.trim.contigs.good.unique.good.good.filter.unique.precluster.pick.
pick.pick.opti_mcc.list,
count=HMWF.trim.contigs.good.unique.good.good.filter.unique.precluster.denovo.vsear
ch.pick.pick.pick.count_table,
taxonomy=HMWF.trim.contigs.good.unique.good.good.filter.unique.precluster.pick.pick.
pick.pds.wang.taxonomy, label=0.03)
```

```
get.oturep(list=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.pick.p  
ick.pick.opti_mcc.list,  
fasta=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.pick.f  
asta,  
count=HMWF.trim.contigs.good.unique.good.filter.unique.precluster.denovo.vsear  
ch.pick.pick.pick.count_table, label=0.03, method=abundance)
```