PLOS Computational Biology

RESEARCH ARTICLE

Panorama: A robust pangenome-based method for predicting and comparing biological systems across species

Jérôme Arnoux*, Jean Mainguy, Laura Bry, Quentin Fernandez de Grado, Yazid Hoblos, David Vallenet, Alexandra Calteau*

LABGeM, Génomique Métabolique, CEA, Genoscope, Institut François Jacob, Université d'Évry, Université Paris-Saclay, CNRS, Évry, France

* arnoux.jeromepj@gmail.com, acalteau@genoscope.cns.fr

OPEN ACCESS

Citation: Arnoux J, Mainguy J, Bry L, Fernandez de Grado Q, Hoblos Y, Vallenet D, et al. (2026) Panorama: A robust pangenome-based method for predicting and comparing biological systems across species. PLoS Comput Biol 22(7): e1013856. https://doi.org/10.1371/journal.pcbi.1013856

Editor: Sreekanth Thota, Vaagdevi College of Pharmacy, INDIA

Received: December 19, 2025

Accepted: June 21, 2026

Published: July 10, 2026

Peer Review History: PLOS recognizes the benefits of transparency in the peer review process; therefore, we enable the publication of all of the content of peer review and author responses alongside final, published articles. The editorial history of this article is available here: https://doi.org/10.1371/journal.pcbi.1013856

Abstract

Over the last decade, the expansion in the number of available genomes has profoundly transformed the study of genetic diversity, evolution, and ecological adaptation in prokaryotes. However, traditional bioinformatic approaches based on the analysis of individual genomes are showing their limitations when faced with the sheer scale of the data. To overcome these constraints, the concept of pangenome has emerged, offering a comprehensive framework to capture the full genetic repertoire of a species. In this study, we present PANORAMA, an innovative pangenomic tool designed to exploit pangenome graphs, enabling their annotation and comparison to explore the genomic diversity of several species. Based on the PPanGGOLiN pangenome graphs, PANORAMA integrates advanced methods for rule-based prediction of macromolecular systems and comparative analysis of conserved features between different pangenomes, such as spots of insertion. We illustrate the use of PANORAMA on a dataset of 941 *Pseudomonas aeruginosa* genomes, evaluating its performance against reference defense system prediction tools such as PADLOC and DefenseFinder. The analysis was then extended to a larger set, including four species of Enterobacteriaceae (>6,000 genomes), demonstrating PANORAMA's ability to annotate, compare, and explore the diversity and distribution of biological systems across multiple species. This work provides new methods for the large-scale comparative study of microbial genomes and highlights the relevance of pangenome approaches in deciphering their evolutionary dynamics. PANORAMA is freely available and accessible at: https://github.com/labgem/PANORAMA

Author summary

Microorganisms are present in nearly all environments on Earth. Uncovering their diversity through the study of their genomes is essential for understanding

Data availability statement: Data Availability All data and code necessary to reproduce the findings of this study are publicly available without restriction. Data The complete dataset, including annotated pangenomes and intermediate analysis files, is available from Zenodo [52] at https://doi.org/10.5281/zenodo.15551255. Raw genomic sequences were obtained from RefSeq; the list of RefSeq accession numbers for all genomes used in this study is provided in the Zenodo repository. Commands to download these genomes programmatically are included in the reproduction notebook. Code for analysis and reproducibility PANORAMA (version 1.0.0) used in this study is available on GitHub at https://github.com/labgem/PANORAMA/tree/1.0.0 under the CeCILL v2.1 license and is archived on Zenodo at https://doi.org/10.5281/zenodo.17877646. A repository containing a Google Colaboratory notebook with all the code to reproduce the figures and analyses presented in this manuscript, including automated downloading of genomic data from RefSeq, is available on GitHub at https://github.com/labgem/PANORAMA_article and is archived on Zenodo at https://doi.org/10.5281/zenodo.17877539. The notebook can be run directly in Google Colaboratory at https://colab.research.google.com/github/labgem/PANORAMA_article/blob/main/main.ipynb.

Funding: This research was supported in part by the CFR PhD program of the French Alternative Energies and Atomic Energy Commission (CEA) for JA and the BlueRemediomics project for JM, which is funded by the European Union under the Horizon Europe Program, Grant Agreement No. 101082304. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

their biology and evolution. This includes characterizing the species' complete genetic repertoire, known as the pangenome. Such research also enables new applications in health, ecology, biotechnology, etc. Here, we present PANORAMA, a novel computational tool designed to predict macromolecular systems, such as defense mechanisms against phages, and to compare pangenome graphs across different species. By using rule-based models that combine gene function and genomic context, we can search for systems directly within pangenome graphs. This graph-based approach provides a global view of the functional content of entire species, moving beyond the analysis of individual genomes. It greatly facilitates the analysis of thousands of genomes by reducing the required computation time and directly integrating the results to identify shared and specific systems. Furthermore, PANORAMA's comparative functionality enables the identification of conserved structures across species, such as shared spots of insertion, revealing common evolutionary mechanisms and functional modules. This work establishes a foundation for comparative pangenomics, offering an unprecedented framework to explore the adaptive potential and evolutionary dynamics of prokaryotes at scale.

Introduction

The rapid expansion of bacterial genome sequencing over the past decade has provided unprecedented opportunities to study the genetic diversity, evolution, and ecological adaptation of microbial organisms [1,2]. For a significant number of species, sequences of hundreds or even thousands of strains are now available. This wealth of information offers immense potential for discovery, but traditional genome-centric approaches, which focus on individual genomes, are increasingly inadequate for managing and interpreting such large-scale datasets. To address these limitations, the concept of pangenome has emerged as a powerful tool. The pangenome encompasses all genomic information within a species or clade, including coding regions, non-coding intergenic sequences, regulatory elements, and mobile genetic elements such as insertion sequences. It provides a holistic framework to capture and compare genomic diversity across strains, distinguishing a *core* component shared by all strains from a *variable* component found only in a subset of them, thereby deepening our understanding of microbial species evolution and adaptation [3]. Pangenomics has significantly transformed microbial genomics by providing a comprehensive framework for understanding genetic diversity and functional capabilities across microbial species [4]. This approach allows researchers to investigate not only the genome of a single strain but also the complete gene repertoire within a species or group of strains, the pangenome, thereby enhancing insights into microbial evolution and adaptation.

Owing to the small size of their genomes and the large number of sequences available, particularly for species of clinical interest, microbial pangenomics has benefited from the early development of tools. These tools facilitate pangenome

construction and analysis and offer capabilities for visualization, comparison, and partitioning of genomic data [5]. Among the computational graph representations developed to analyze pangenomes, sequence-level graphs — such as De Bruijn graphs [6] and variation graphs [7] — encode genomic variation at the nucleotide level, while gene-level adjacency graphs represent gene family presence/absence and genomic contiguity across strains. Among the latter, PPanGGOLiN stands out for its unique approach to analyzing pangenomes by partitioning them with a statistical algorithm [8]. PPanGGOLiN represents genomic data as a gene adjacency graph at the gene family level, with nodes representing homologous gene families and edges capturing their genomic contiguity, enabling the compression of information from thousands of genomes while preserving the chromosomal organization of genes. A statistical model is applied to partition gene families into *persistent* genome (*i.e.*, gene families conserved in nearly all genomes) and variable genome, which includes the *shell* and *cloud* components corresponding to intermediate- and low-frequency gene families, respectively. PPanGGOLiN includes additional methods for the identification of Regions of Genomic Plasticity (RGPs) and their spot of insertion (panRGP method) [9] as well as their fine description in conserved modules (panModule method) [10]. These methods have demonstrated their utility in identifying genomic islands and have provided valuable insights into the genomic adaptability and evolution of bacteria.

Despite these advances, a significant challenge remains: detecting and comparing complex macromolecular systems at the pangenome scale. In microbial genomes, these systems correspond to multi-protein assemblies encoded by genes usually arranged in a highly structured manner, typically clustered into one or several operons composed of functionally related genes. Their strong genomic co-occurrence and co-localization make them amenable to systematic detection; tools such as Coinfinder [11], Goldfinder [12], and panModule [10] exploit this property to identify conserved gene clusters across a set of genomes. Biologically, these clusters encode coordinated systems that play essential roles in microbial life. Among them are secretion systems, which allow the transport of proteins and other molecules across membranes to interact with the environment or host organisms; defense systems, which protect the cell from foreign genetic elements; and metabolic pathways, which organize enzymatic reactions to efficiently produce, transform, or degrade biological molecules. Understanding the organization and diversity of these systems is key to decoding the functional capabilities of microbial genomes. Several tools have been developed to detect macromolecular systems at the genome scale. For example, Hamburger [13] identifies spatially constrained subsets of proteins within a defined genomic neighborhood and has been applied to the detection of Type VI secretion systems. As for Spacedust [14], it follows a different strategy, relying on protein structure comparisons and ordered conservation matrices; it has been applied, for example, to the detection of defense systems. In contrast, other tools define system-specific rule-based models to improve detection accuracy and specificity. Among them are MacSyFinder [15], PADLOC [16], and DefenseFinder [17]. The latter two are specialized in identifying bacterial anti-phage immune systems. They are highly effective when applied to individual genomes but are not designed to support systematic comparisons across large genomic datasets, such as detecting systems directly at the pangenome level. Tools capable of functionally annotating a pangenome at the scale of thousands of genomes remain limited. While some approaches can incorporate external annotations as additional metadata, such as PPanGGOLiN [8], Panaroo [18] and PanTOOLs [19], they do not support direct functional annotation of pangenomes, relying instead on predictions generally made at the genome level. Moreover, no tool enables the comparison of functional annotations across pangenomes from multiple species.

Here, we introduce PANORAMA, a powerful computational tool designed to harness bacterial gene-level pangenome graphs from large genomic datasets and enable comparisons across species to explore genomic diversity. Built on the PPanGGOLiN software suite, PANORAMA incorporates advanced methods for reconstructing and analyzing pangenome graphs. It offers several key features, including the ability to compare genomic contexts between pangenomes and annotate macromolecular systems at the pangenome scale. Functional annotation of biological systems is performed directly on the graph structures using rule-based models, making it possible to map and analyze complex genomic features without relying on linear genome representations. To illustrate the versatility of our approach, we focused on the comparative

analysis and annotation of bacterial defense systems. Bacteria have evolved a remarkably diverse array of defense mechanisms against phages and other mobile genetic elements. These range from well-characterized systems, such as restriction-modification (RM) systems [20] and CRISPR-Cas complexes [21], to more recently discovered and less understood systems like BREX [22], DISARM [23], and retron-based defense systems [24,25]. To date, over 150 systems have been described, unveiling an unsuspected diversity of molecular mechanisms [26]. This diversity is not only taxonomically widespread but also highly dynamic; defense systems are often associated with mobile genetic elements or genomic islands and can vary extensively in composition, organization, and presence both within and between closely related species [17,27]. Despite this complexity, large-scale comparative studies of bacterial defense systems are still scarce. Pangenome-level analyses hold great promise for revealing patterns of co-occurrence, horizontal gene transfer, evolutionary innovation, and defense strategies that are specific to particular species or lineages [28]. In this study, we present the methodology behind PANORAMA, a graph-based pangenomic framework designed to address these challenges. We first demonstrate its application on a comprehensive dataset of *Pseudomonas aeruginosa* genomes and compare its performance to established genome-scale tools such as PADLOC and DefenseFinder. We then extend this analysis to a broader dataset comprising four Enterobacteriaceae species, showcasing PANORAMA's capacity to annotate, compare, and explore the distribution and diversity of phage defense systems across species. Overall, this work provides a powerful new resource for the comparative study of bacterial genomes and highlights the value of pangenomic approaches in revealing the evolutionary dynamics and ecological significance of bacterial defense repertoires.

Results and discussion

Overview of PANORAMA

To predict macromolecular systems, PANORAMA employs rule-based models similar to those used in MacSyFinder [15]. However, instead of applying these models to individual genomes, PANORAMA operates on the pangenome graph structure of PPanGGOLiN [8]. The rules rely on the presence/absence of specific functions predicted from pangenome gene families, incorporating constraints on their genomic organization (*i.e.*, gene colocalization). Functional annotation of pangenome graph gene families is performed through alignments with HMM protein profiles [29] defined for each macromolecular system. The genomic contiguity of gene families potentially involved in a system is then assessed on the pangenome graph by applying transitive closure and edge filtering. Finally, the predicted systems consist of sets of colocalized gene families from the pangenome graph, supplemented with information on their classification within the *persistent*, *shell*, or *cloud* genome, as well as their association with RGPs, modules, and spots of insertion. Systems are also projected onto the genomes to determine their presence and gene content in each strain. An additional functionality of PANORAMA is its ability to compare pangenomes, identifying similar systems and spots of insertion across species. Based on a set of predicted spots or systems in several pangenomes, PANORAMA computes a similarity score for each pair of elements by detecting shared gene families. Then, it applies a community clustering algorithm to group similar systems or spots into clusters. More details about PANORAMA methods are provided in the Materials and Methods section.

PANORAMA is available as open-source software written in Python and is designed for easy installation and usage to facilitate broader adoption by the community (https://github.com/labgem/PANORAMA). Software commands are organized into two main workflows (Fig 1). The PanSystem workflow begins by annotating gene families of the pangenome graph using the specified HMM library, and then applies system prediction rules from the model repository. Gene family annotations can also be performed externally and provided to PANORAMA by the user in a Tab-Separated Values (TSV) file. The PanCompare workflow performs comparative analyses of two or more pangenomes, including gene family clustering and system/spot comparisons. Both workflows generate textual outputs (TSV files), graph-based representations (in GEXF or GraphML formats, compatible with Gephi [30] or Cytoscape [31] software for visualization), and figures to summarize results. Functional annotations and predicted systems are saved in the pangenome's HDF5 file, allowing for further analysis. Additional utilities are provided to automatically convert system models and HMM libraries into the PANORAMA

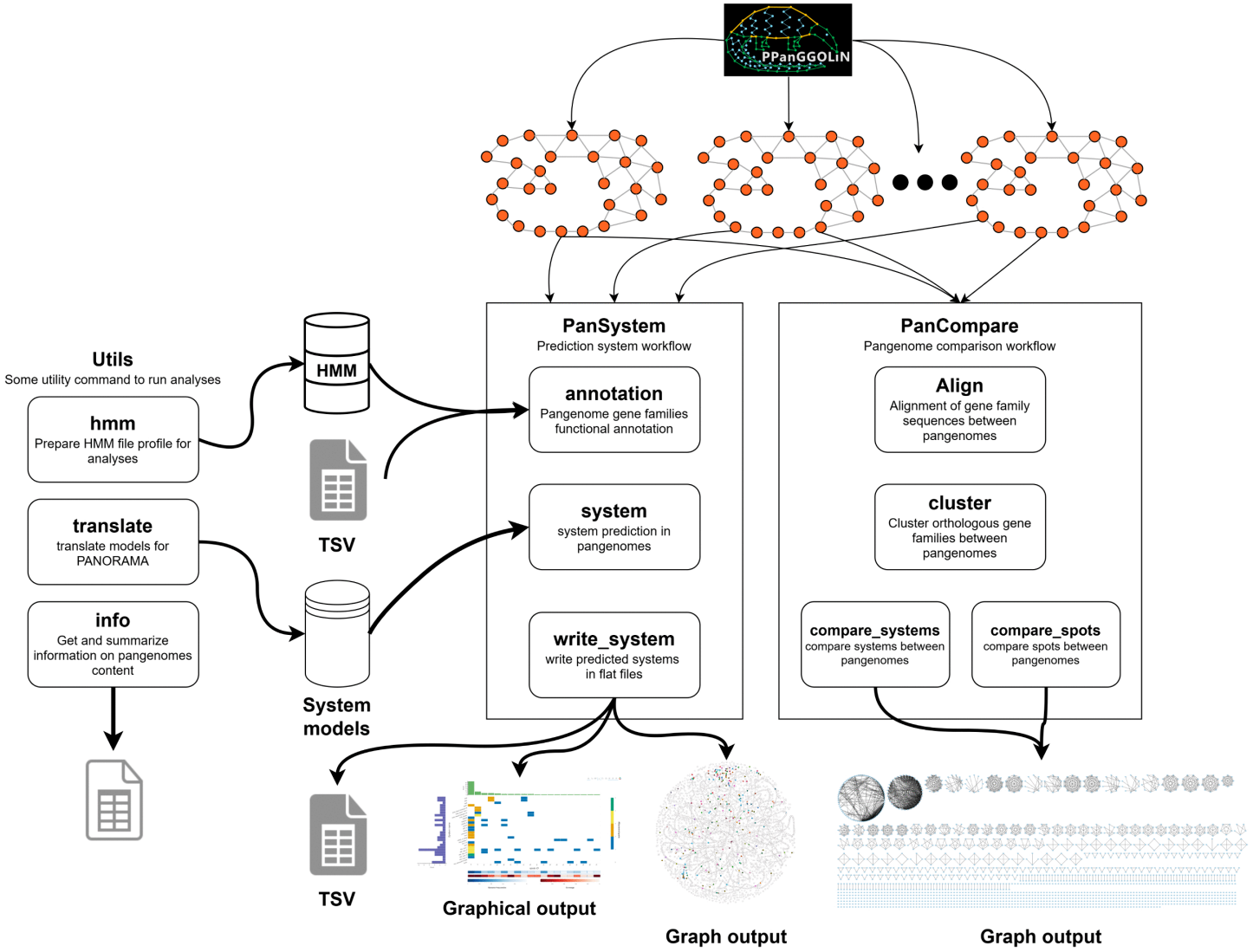

Fig 1. PANORAMA software overview. Each rounded box represents a possible software command integrated into workflows depicted in square boxes. PANORAMA is organized into two main workflows: *PanSystem*, which focuses on system prediction and annotation within pangenomes, and *PanCompare*, which handles comparative analyses of pangenomes, including gene family alignment, clustering, and system/spot comparisons. Utility tools, such as HMM preparation and information summarization, facilitate data input and model translation. The software outputs include TSV files, graphical visualizations, and graph-based representations for comparing systems and pangenomes.

https://doi.org/10.1371/journal.pcbi.1013856.g001

format, with support for models from MacSyFinder [15], DefenseFinder [17], CasFinder [32], and PADLOC [16]. Models are stored in JavaScript Object Notation (JSON) format, with a flexible and easy-to-understand grammar, enabling users to customize or create new models. The complete creation workflow and available model collections are described in the Materials and Methods section.

System prediction benchmark. A set of 941 complete genomes of *Pseudomonas aeruginosa* was used to evaluate PANORAMA's defense system predictions against two reference tools, DefenseFinder (including CasFinder models) and PADLOC. Although these tools use a similar approach to predict defense systems, they differ in the number of models (281 in DefenseFinder vs. 385 in PADLOC) and in the parameters used for predicting functions from HMM alignments, as

well as for applying presence/absence and colocalization rules. Thanks to its generic system representation, PANORAMA is compatible with both tools and was run using their respective system models and HMMs after format conversion (*i.e.*, PanSystem workflow).

To conduct this benchmark, we assessed whether PANORAMA correctly assigned pangenome gene families to the appropriate systems based on the results from DefenseFinder or PADLOC. As expected, we obtained highly similar results, achieving an F1-score of 98.01% (recall: 98.84%, precision: 97.33%) using PADLOC as a reference and 96.64% (recall: 98.23%, precision: 95.10%) with DefenseFinder. As shown in Fig 2A, a substantial number of families are shared exclusively between PANORAMA and either DefenseFinder (650 families) or PADLOC (855 families), while only 964 families are common. This highlights the complementary nature of these tools and demonstrates that the pangenome analysis provided by PANORAMA is highly valuable for reconciling their results. Besides, PANORAMA missed only a few families: 56 for DefenseFinder and 47 for PADLOC (Table 1). These cases correspond to missing annotations in pangenome gene families. Unlike PADLOC and DefenseFinder, which analyze each sequence individually for every genome, PANORAMA relies on the protein sequence of the family's representative gene for HMM alignments. Consequently, in rare instances, the alignment of the representative sequence falls just below the defined threshold, even though other sequences in the family have hits above it. Conversely, PANORAMA can identify additional results (318 families) for certain genomes whose gene sequence alignment with the HMM is less conserved than with the representative sequence. A final point concerns

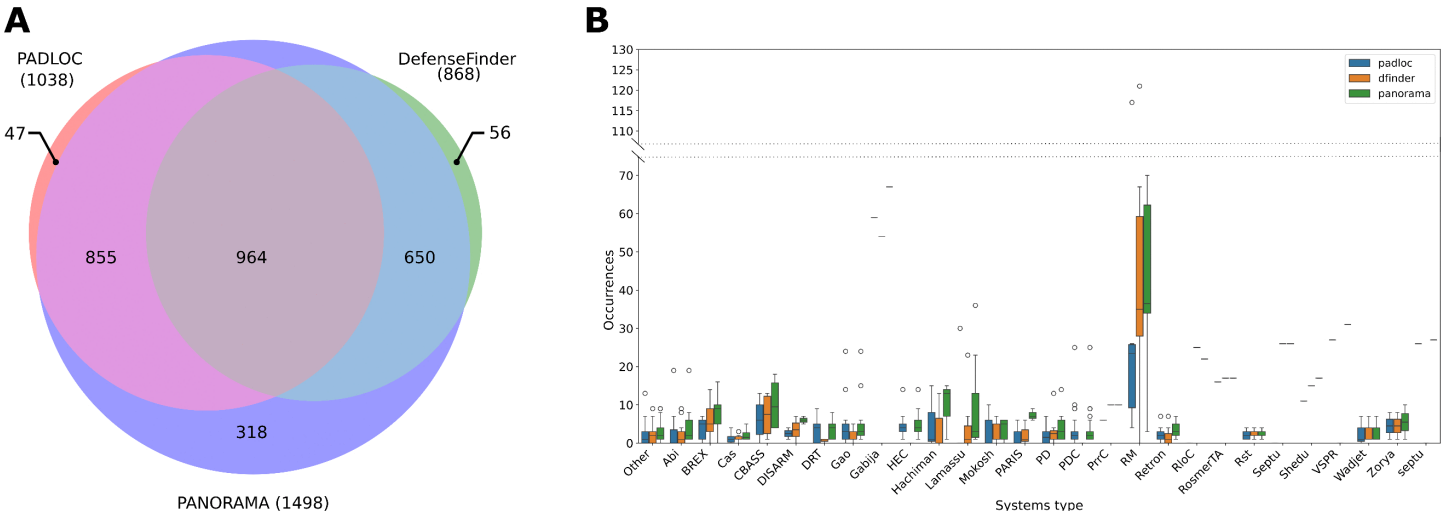

Fig 2. Comparison of PANORAMA, PADLOC, and DefenseFinder system predictions at the pangenome level. PADLOC and DefenseFinder predictions at the genome level were unified at the pangenome level by converting sets of system genes to sets of gene families. **(A)** Venn diagram illustrating the overlap of system families predicted by the three tools. **(B)** Boxplots displaying the distribution of system counts for each category, as predicted by the different tools.

https://doi.org/10.1371/journal.pcbi.1013856.g002

Table 1. System prediction comparison between PANORAMA, PADLOC and DefenseFinder. For each comparison, M1 refers to the first method listed (Method 1) and M2 to the second method (Method 2). Common M1 and Common M2 indicate the number of systems predicted by both methods and attributed to M1 and M2 respectively. Specific M1 and Specific M2 represent systems uniquely predicted by each method. Numbers in parentheses indicate the total number of systems predicted by each method.

Method 1 / Method 2	Common M1	Common M2	% common	Specific M1	Specific M2
PADLOC (1038) / DefenseFinder (868)	489	493	51.52%	549	375
PADLOC (1038) / PANORAMA (1064)	1003	992	94.91%	35	72
DFinder (868) / PANORAMA (949)	828	813	90.31%	40	136

https://doi.org/10.1371/journal.pcbi.1013856.t001

the systems predicted only by PANORAMA (*i.e.*, 136 with DefenseFinder models and 72 with those of PADLOC, see Table 1). Although such a high number was not initially expected, a detailed analysis revealed that some of these additional systems arise because PANORAMA's genomic context search, based on colocalization rules in the pangenome graph, is less stringent than that of PADLOC and DefenseFinder. This relaxed approach allows PANORAMA to detect additional genes that are consistently conserved alongside known system components (see Materials and methods for further explanation). Other additional predictions stem from differences in HMM annotation due to the use of representative sequences for families, as mentioned earlier, but also from a bug in DefenseFinder that affected only RM and Retron systems, where alignment thresholds were not properly handled by the software. This bug has since been corrected by the authors (version 2.0.0), though it was not tested in this study. Other discrepancies arise because PANORAMA allows multiple associations between gene families and systems, whereas PADLOC and DefenseFinder link each gene to only one system. This is especially noticeable for systems with closely related models; for example, PANORAMA often predicts multiple Retron, RM, or Lamassu systems when other tools identify only one (Fig 2B). One potential improvement would be to include *forbidden* families in the models to facilitate their differentiation or to implement a scoring function, as done in MacSyFinder, to identify the best system. PANORAMA may also predict additional systems at the genome level for which the colocalization rule is not respected. These systems, flagged as *split*, have their genes separated, possibly due to rearrangements, insertion events, or assembly breaks occurring in a subset of genomes in which they are predicted.

Tool execution performance was also evaluated by measuring their runtime and memory usage on a Linux server with 36 CPU cores (Table 2). For a fair benchmark, all tools were executed sequentially using a single CPU core: PADLOC and DefenseFinder processed each genome individually in succession, while PANORAMA first loaded the entire pangenome before sequentially predicting defense systems. PPanGGOLiN pangenome construction time is reported separately as a prerequisite step for PANORAMA; it is run once and its output can be reused across multiple analyses (S1 Table). Despite this serial execution, PANORAMA significantly outperforms the other tools in runtime, completing analyses in minutes (S1 Fig & S1 Table) compared to 24 and 71 hours for DefenseFinder and PADLOC, respectively. This efficiency is achieved by analyzing gene families rather than individual genes, which reduces computational overhead. Additionally, PANORAMA utilizes pyHMMER [33] for HMM alignments, optimizing the workflow by minimizing I/O operations and enabling on-the-fly result filtering. In contrast, other tools use the HMMER software directly [29], which requires post-processing steps.

Pseudomonas aeruginosa defense system analysis

System prediction and analysis. An exhaustive analysis was conducted on PANORAMA's defense system predictions, utilizing the DefenseFinder models and the same set of *P. aeruginosa* genomes as for the benchmark. PANORAMA identified a total of 949 systems in the pangenome, categorized into 150 distinct models, with restriction-modification (RM) systems being the most abundant (Fig 3). RM systems are present in 84% of genomes, with nearly 300 detected, of which type I systems are the most common, occurring 130 times. The Gabija, CRISPR-Cas, and CBASS system categories follow as the next most prevalent defense systems in genomes, each with a presence rate above 40%.

Table 2. Execution performance on *P. aeruginosa* pangenome. Performance comparison of defense system prediction tools on the *P. aeruginosa* pangenome made of 941 genomes, grouped by reference database. All tools were run sequentially on a single CPU with dedicated machine resources. PANORAMA timing excludes pangenome construction.

Tool (version)	Database (version)	#Systems predicted	Run Time (hh:min)	Peak Memory (GB)
DefenseFinder (1.2.2)	Defense-finder-models (1.2.4) & CasFinder (3.1.0)	868	24:45:22	6.84
PANORAMA (1.0.0)		949	00:20:05	11.22
PADLOC (2.0.0)	PADLOC-DB (2.0.0)	1038	71:27:43	3.19
PANORAMA (1.0.0)		1064	00:33:33	10.77

https://doi.org/10.1371/journal.pcbi.1013856.t002

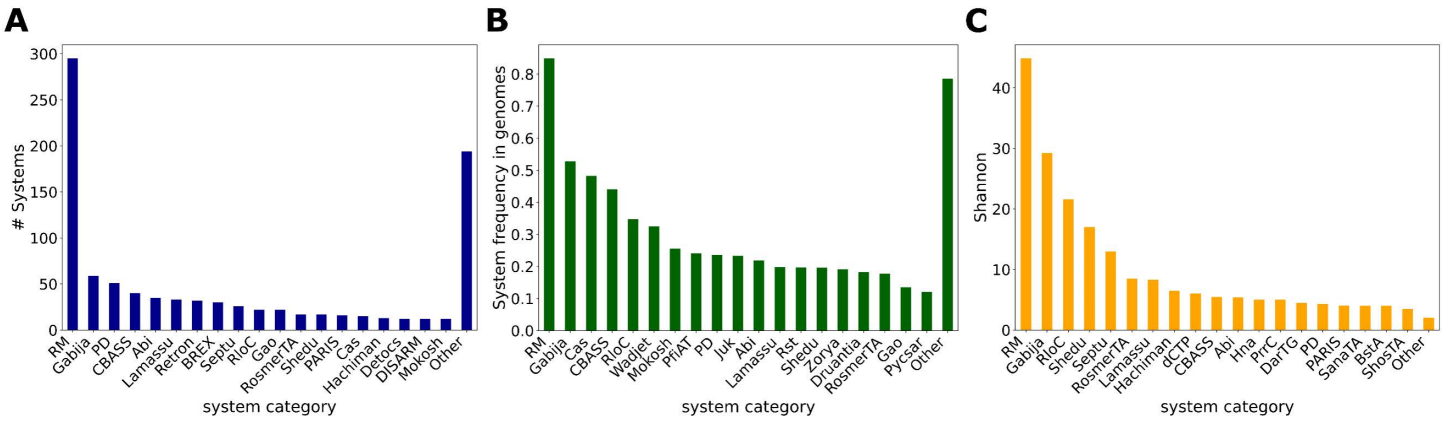

Fig 3. System prediction metrics in *P. aeruginosa*. Systems are grouped by categories on the x-axis and ordered by decreasing values. Only the 19 highest-value system categories are displayed, with others grouped under the "Other" category. **(A)** Number of systems found for each category in the pangenome. **(B)** Relative frequency of system categories in genomes. **(C)** Shannon entropy of system categories.

https://doi.org/10.1371/journal.pcbi.1013856.g003

These observations corroborate the finding of Johnson *et al.* [34]. At the pangenome level, some system categories are highly prevalent across genomes but are represented by only a few distinct systems. For example, CRISPR-Cas systems appear in 48% of genomes, yet only 13 distinct systems are identified in the pangenome. This is further highlighted by a Shannon entropy calculation, which measures the compositional diversity of system categories (Fig 3C). For CRISPR-Cas systems, the entropy is 1.35, indicating a high degree of conservation in the gene family composition across genomes. Among the most prevalent system categories, others show notable diversity, including RM, Gabija, and RloC. The RloC systems, for example, consist of 22 distinct systems spread across 35% of genomes, with a Shannon entropy of 21.58, indicating considerable variability in their family composition. These predictions are consistent with recent studies and highlight the remarkable diversity of anti-phage immune systems in prokaryotes [17].

Defense islands and spots of insertion. PANORAMA systems can be analyzed in conjunction with additional information extracted from the PPanGGOLiN [8] pangenome graph, particularly concerning their association with the variable genome and their localization within RGPs and spots of insertion predicted by panRGP [9]. This enables the identification of defense islands (*i.e.*, variable regions enriched with defense systems) and their hotspots (*i.e.*, frequently occurring insertion sites of defense islands in genomes).

Most defense systems are predicted within the variable (*shell* or *cloud*) genome of *P. aeruginosa* and are located in spots. PANORAMA identified 395 spots containing at least one defense system, representing 40% of all pangenome spots. Among them, four spots (7, 6, 45, 69) have a high frequency (>25%) and exhibit the highest number of defense systems predicted at the pangenome level (>60 systems) (Fig 4). Notably, spots 7 and 6 are the most diverse, harboring 238 and 162 associated systems, respectively (Fig 5). They are mostly composed of RM systems (51% and 56%) but also exhibit a broad diversity of other categories, including BREX (6%), Gabija (5% and 4%), PD (4% in spot 7) and CBASS (4% in spot 6). These two spots were previously identified using a non-automated approach in the study by Johnson *et al.* [34] as core defense hotspots in *P. aeruginosa*, where they were designated CDHS-1 and CDHS-2. This further highlights the value and reliability of PANORAMA in automatically detecting defense islands and their spots. Using PANORAMA, we also identified two additional defense hotspots (spots 45 and 69). Spot 45 contains 110 systems and stands out as the most balanced in terms of system categories; it is also the only hotspot with a notable presence of PARIS systems (8%). Spot 69 contains 75 systems and is dominated by RM systems (like spots 7 and 6), with PrrC systems specifically represented at 7%. Although less frequently observed across genomes (<20%), spots 61 and 1 display highly diversified system content, comprising 72 and 65 systems, respectively. Both are also rich in RM systems, with Mokosh

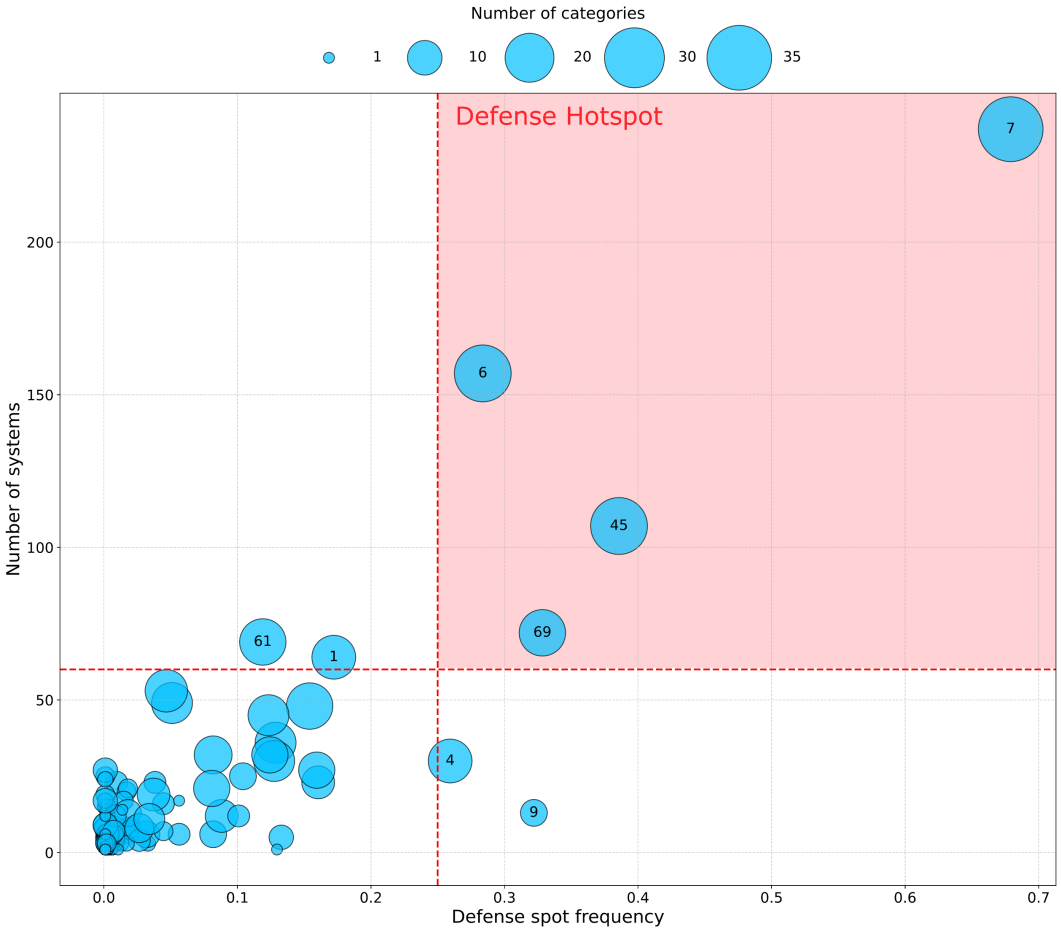

Fig 4. System diversity and defense spot frequency in *P. aeruginosa*. This bubble plot displays the distribution of defense spots identified by PANORAMA, based on their frequency in genomes (x-axis) and the total number of defense systems identified within each spot at the pangenome level (y-axis). The size of the bubbles is proportional to the number of distinct system categories represented in each spot (legend at top shows scale), and the numbers within bubbles indicate spot identifiers.

https://doi.org/10.1371/journal.pcbi.1013856.g004

particularly represented in spot 1 and CBASS and PD systems notably present in spot 61 (≃8%). Finally, spots 4 and 9 are relatively frequent across genomes (≃30%) but contain few distinct systems, 31 and 10, respectively. These results highlight the potential of PANORAMA to provide a comprehensive landscape of defense systems in a species, enabling pangenome-scale analysis and the identification of defense islands along with their hotspots of insertion.

Pangenome comparison of Enterobacteriaceae defense arsenal

To demonstrate the comparative functionalities of PANORAMA (*i.e.*, PanCompare workflow), the pangenomes of four Enterobacteriaceae species were analyzed for defense system and spot prediction. This dataset represents more than 6,000 genomes from *Citrobacter freundii*, *Salmonella enterica*, *Klebsiella pneumoniae*, and *Escherichia coli* species (Table 3). The distribution of systems among species and their association with conserved spots were studied. Defense systems were predicted using DefenseFinder models.

Defense system distribution in the four species. A total of 344, 452, 988, and 1470 defense systems were predicted from the pangenomes of *C. freundii*, *S. enterica*, *K. pneumoniae*, and *E. coli*, respectively. In addition to

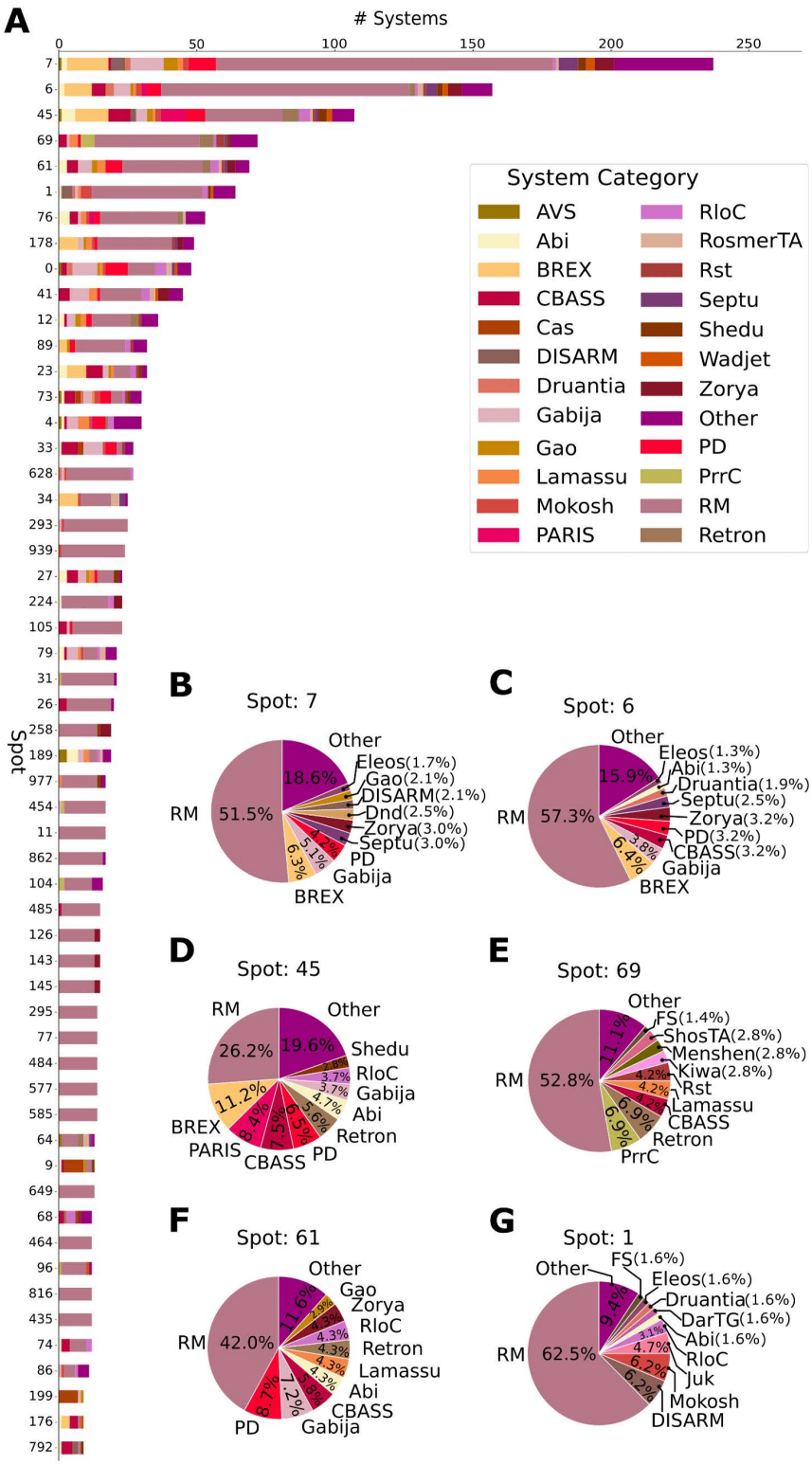

Fig 5. Defense systems within insertion spots of the *P. aeruginosa* pangenome. (A) The bar plot displays the number of predicted defense systems in the *P. aeruginosa* pangenome for each spot. Only spots with at least 10 systems are displayed. **(B-G)** The pie charts illustrate the system category composition for the six major spots.

https://doi.org/10.1371/journal.pcbi.1013856.g005

Table 3. Pangenome statistics across bacterial species. Summary statistics for five bacterial species analyzed in this study.

Species	Genomes	Genes	Families	Edges	Persistent	Shell	Cloud	RGPs	Spots
C. freundii	79	385611	16865	24936	3780	2954	10131	3863	254
E. coli	3083	14447407	44278	140743	3018	7174	34086	240163	2036
K. pneumoniae	1659	8851686	32685	75073	4155	5085	23445	67678	778
P. aeruginosa	941	5811655	34058	61841	4925	6768	22365	44935	984
S. enterica	1380	6222501	23941	46535	3474	4080	16387	65920	458

https://doi.org/10.1371/journal.pcbi.1013856.t003

textual outputs, PANORAMA automatically generates a heatmap that displays system occurrences across the compared pangenomes (see S2 Fig). Among the different categories, RM systems are the most abundant in the four species, accounting for 30% to 45% of the systems found. Following RM systems, PD systems are the next most prevalent, representing 5% to 7% of the systems within Enterobacteriaceae (Fig 6). Other notable system categories, such as Retron, CBASS, Gao, BREX, and Abi, are also relatively abundant across all pangenomes. Next, we evaluated the species-specificity of each system category by computing enrichment factors (S3 Fig). Our findings indicate that certain categories, Azaca and Dnd in *S. enterica*, Juk in *K. pneumoniae*, and Bunzi and RADAR in *C. freundii*, exhibit enrichment factors above 3, highlighting their preferential association with these species. Tiamat systems are only found in *S. enterica* and *C. freundii*. Interestingly, *E. coli* does not show any systems in higher abundance compared to other species. Beyond the large number of genomes used to construct the pangenome, this pattern may reflect that the *E. coli* dataset includes strains exposed to a wider diversity of phages, a consequence of its commensal lifestyle in the gut. This observation warrants further investigation by conducting genome dereplication or rarefaction analysis to evaluate whether this pattern remains robust in subsets of genomes.

Identification of conserved spots. We identified 150, 291, 635, and 1664 spots with at least one defense system in *C. freundii*, *S. enterica*, *K. pneumoniae*, and *E. coli*, respectively. The most defense system-rich spot in our analysis is spot 35 in *E. coli* (S4 Fig), which contains more than 450 systems and corresponds to the tRNA-LeuX genomic island. This island, previously characterized for harboring multidrug resistance genes [35], is revealed here to also serve as a major phage defense hotspot, demonstrating the dual protective role of this genomic region.

With PANORAMA, we searched for similar spots based on their related gene families, using a gene family repertoire relatedness (GFRR, see Pangenome comparison workflow) threshold of at least 60%. This threshold guarantees at least one similar gene family on each side of the border. As shown in Fig 7, we identified 102 clusters of similar spots, encompassing 232 spots that each share at least one neighboring gene family composition with another spot across the Enterobacteriaceae pangenomes.

E. coli is the species that shares the most spots with others (71), followed by *S. enterica* (60), then *C. freundii* (48), and *K. pneumoniae* (46). Proportionally to its number of spots, *C. freundii* is the species with the most similar spots, with 18% of its spots similar to the other pangenomes. The two species with the most common spots are *E. coli* and *S. enterica*, with 67 common spots. PANORAMA can then be used to highlight spots of insertion at a higher taxonomic rank than the species. These could be areas of interest for research into shared evolution or exchanges between these species.

Defense spot identification and conservation. Using the comparative functionalities of PANORAMA, we identified 99 clusters of similar spots conserved in at least two of the four Enterobacteriaceae species (Fig 7). As might be expected, given their phylogenetic proximity, *E. coli* and *S. enterica* share the most common spots (*i.e.*, 34 spot clusters, 26 of which are found only in these two species). About half of the spot clusters (n = 47) are associated with a defense system in at least one species, comprising a total of 520 distinct defense systems among the 3,265 identified in the four species' pangenomes. Of these, only one spot cluster (cluster 58), containing 34 defense systems, is conserved in all pangenomes.

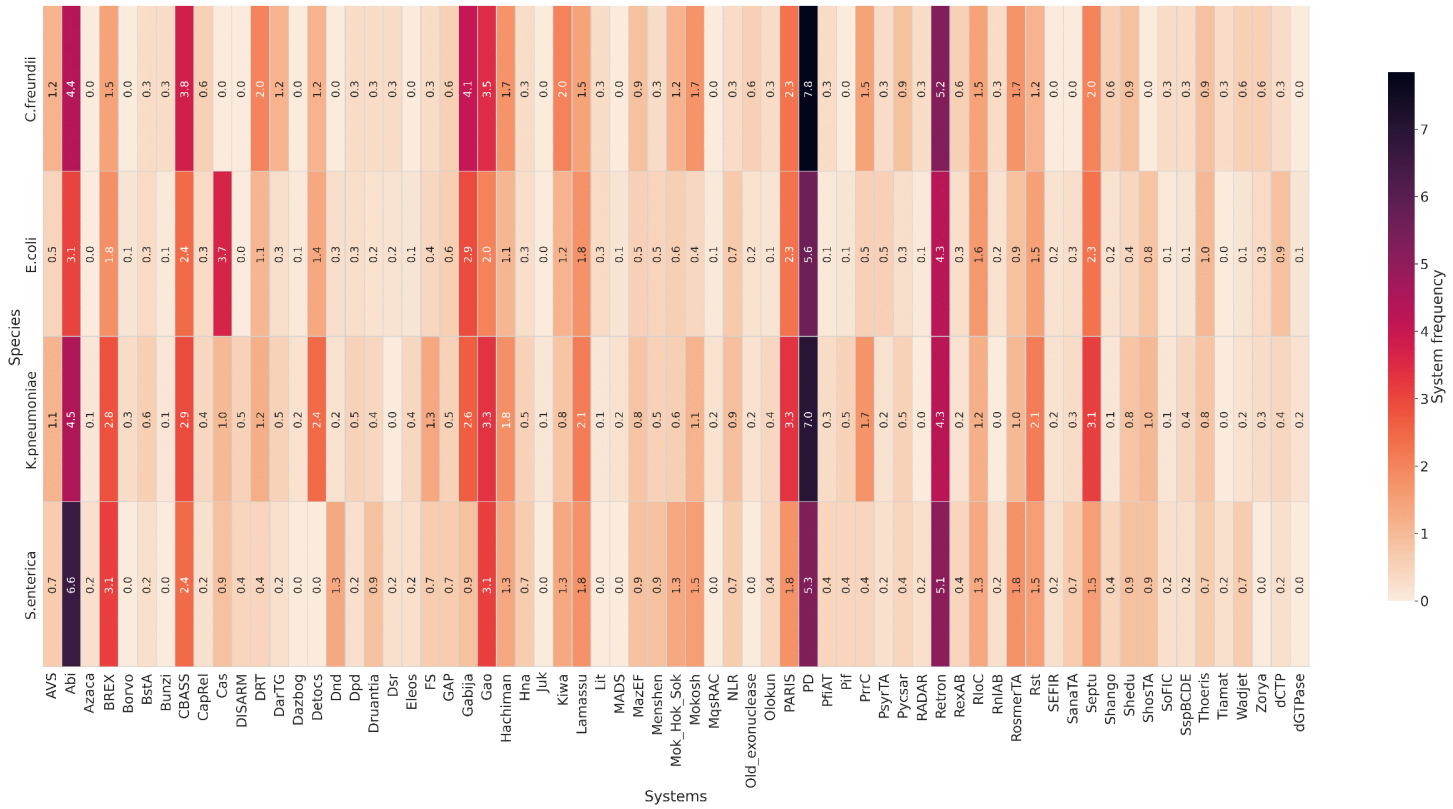

Fig 6. Relative frequency of system categories in Enterobacteriaceae pangenomes. The relative frequencies are expressed as a percentage. The RM system category was omitted to facilitate interpretation.

https://doi.org/10.1371/journal.pcbi.1013856.g006

E. coli and *S. enterica* harbor the highest number of defense systems (263 systems) across their specific spot clusters (7 clusters). One of these clusters (spot cluster 3062) includes spot 86 in *S. enterica* and spot 175 in *E. coli*. These spots rank, respectively, eighth and sixth in terms of the number of defense systems (S4 Fig), with 32 in *S. enterica* and 216 in *E. coli*, and can therefore be considered as potential defense islands or even defense hotspots. Analyzing their composition reveals notable similarities (S5 Fig). Both spots are mainly composed of RM systems (55% in *S. enterica*, 85% in *E. coli*), followed by BREX, CBASS, and PrrC, which are generally not phage-specific. These findings support the hypothesis that defense systems may have been exchanged between the two species from this conserved hotspot. Interestingly, spot 175 in *E. coli* corresponds to a previously identified defense hotspot (spot 4) by Hochhauser *et al.* [28]. The expanded system diversity and abundance we observe, compared to the previous study, reflect our larger genome dataset.

Examining the system category diversity of other spot clusters (Flg 8), many clusters are predominantly composed of RM systems. Some clusters are dedicated to a single system category, such as BstA in cluster 2320, while others, like the previously mentioned cluster 58, are more diverse, bringing together systems from all four species studied.

Beyond their illustrative purpose, the analyses presented here highlight the ability of PANORAMA's comparative functionalities to identify conserved defense islands across species, providing valuable insights into the evolution of defense systems and their mechanisms of acquisition.

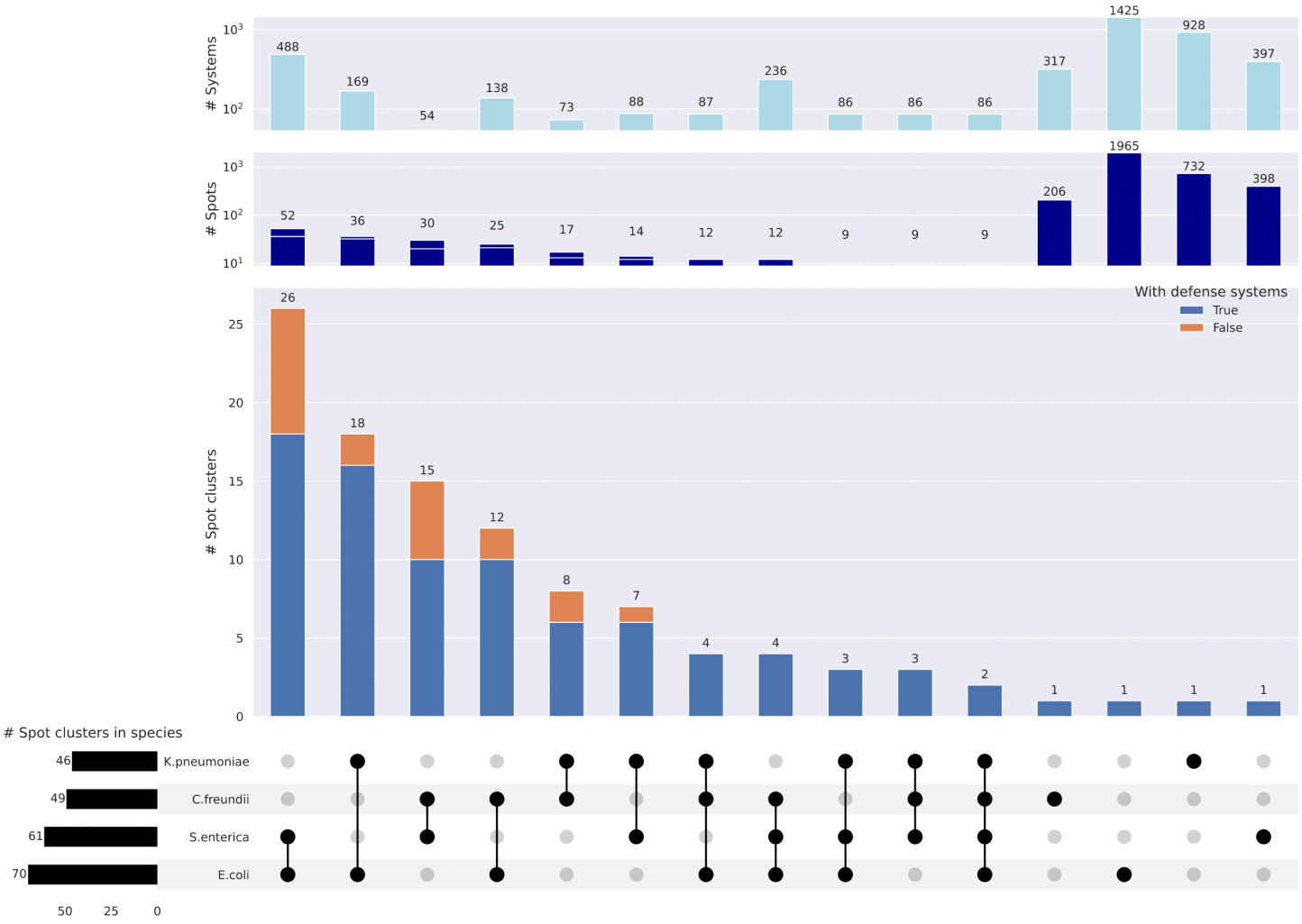

Fig 7. Sharing of spot clusters across four Enterobacteriaceae species and their association with defense systems. The UpSet plot shows the number of spot clusters shared across the four compared species, with stacked bars to indicate whether they contain defense systems (in blue) or not (in orange). The two top bar plots represent spot and defense system abundance metrics within spot clusters on a logarithmic scale.

https://doi.org/10.1371/journal.pcbi.1013856.g007

Conclusion

In this work, we introduced PANORAMA, a novel open-source framework for the prediction and analysis of macromolecular systems across prokaryotic pangenomes. Unlike existing tools that operate on individual genomes, PANORAMA leverages the pangenome graph structure provided by PPanGGOLiN to perform system-level analyses at both species and interspecies scales. It utilizes rule-based models inspired by those of MacSyFinder, but adapted to pangenomes, enabling the flexible and accurate identification of diverse functional systems, such as defense mechanisms.

Benchmarking against established tools, such as DefenseFinder and PADLOC, demonstrated that PANORAMA achieves comparable prediction accuracy while reducing computational demands, making it particularly well-suited for large-scale analyses involving hundreds or thousands of genomes.

Beyond its performance, PANORAMA introduces key methodological innovations. Notably, its graph-based approach enables the detection of conserved genomic contexts by identifying gene families that consistently co-occur across

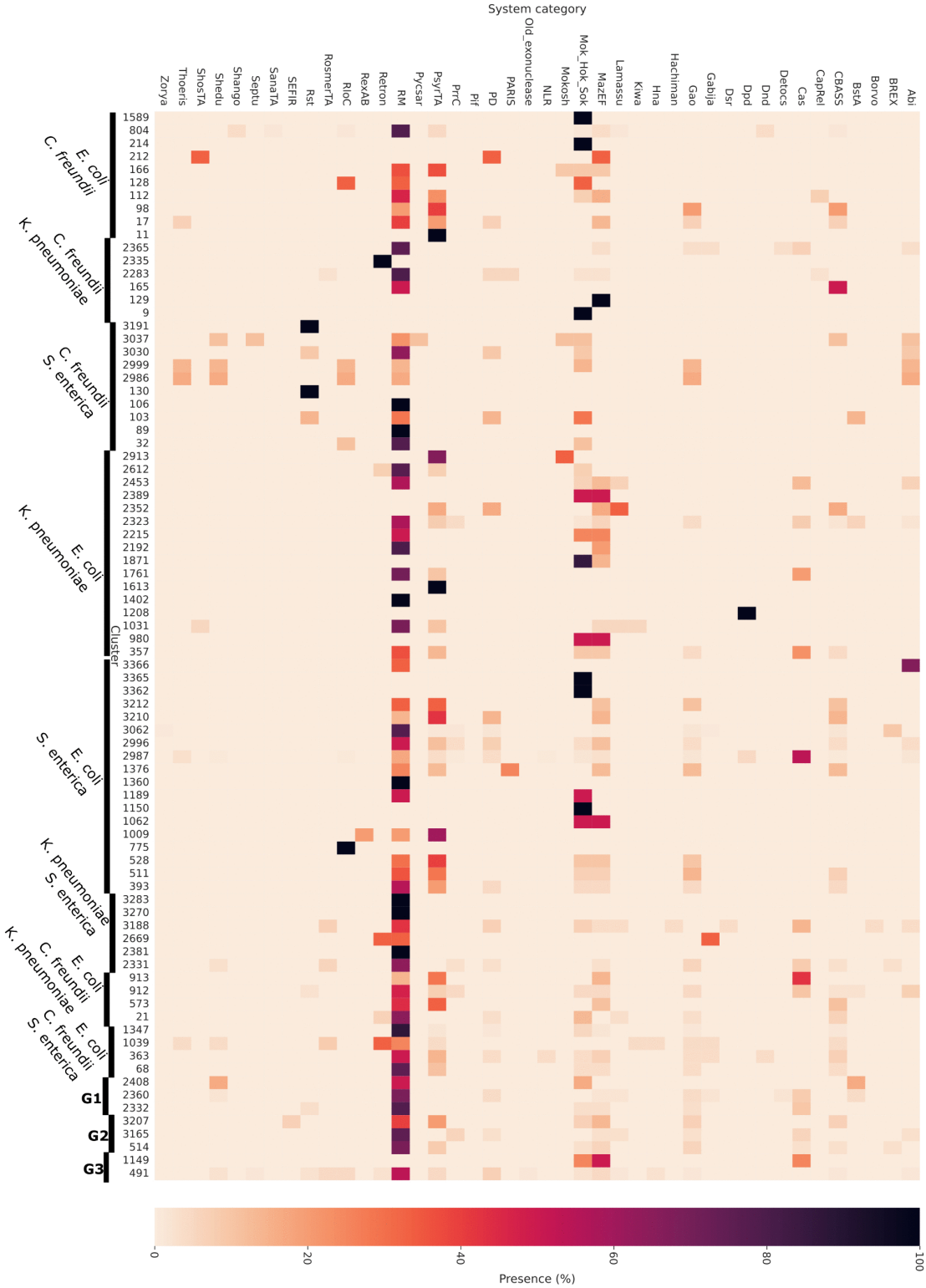

Fig 8. Relative frequency of system categories in each spot cluster. The heatmap presents the relative frequency (%) of system categories within each spot cluster. Species associated with each spot cluster are indicated next to the corresponding cluster identifiers. G1: *C. freundii*, *K. pneumoniae*, *S. enterica*; G2: *E. coli*, *K. pneumoniae*, *S. enterica*; G3: *C. freundii*, *E. coli*, *K. pneumoniae*, *S. enterica.*

https://doi.org/10.1371/journal.pcbi.1013856.g008

multiple genomes. This strategy allows for the robust identification of evolutionarily maintained system components, even when their genomic proximity is disrupted in some genomes by rearrangements, insertions, or assembly fragmentation. Consequently, PANORAMA offers greater flexibility than traditional genome-based tools in identifying atypical genomic architectures.

Applied to hundreds of genomes of a species, PANORAMA can quickly identify the different systems and their occurrences in individual genomes. A notable strength lies in its ability to associate systems with regions of genomic plasticity and hotspots of insertion, allowing for the identification of genomic islands enriched with systems and their preferred integration sites. In *P. aeruginosa*, four major hotspots were identified, including two that correspond to previously published core defense hotspots.

To further illustrate the comparative functionality of PANORAMA, we analyzed the defense repertoires of over 6,000 genomes from four Enterobacteriaceae species. PANORAMA revealed both conserved and species-specific system categories and was able to cluster insertion spots based on shared gene family content. This enabled the identification of conserved defense islands across phylogenetically related species, offering insights into the evolutionary conservation of these systems.

Altogether, PANORAMA provides a robust and extensible framework for system detection and comparative analysis at the pangenome level. An important advantage of PANORAMA is that its predictions are inherently produced at the pangenome scale: whereas genome-based tools require the aggregation and harmonization of hundreds or thousands of individual genome-level outputs to obtain species-wide summaries, PANORAMA directly provides unified results through its graph-based representation. Its flexible modeling format, compatibility with existing system model databases, and integrative workflows make it a valuable tool for microbiologists, bioinformaticians, and evolutionary biologists interested in understanding the distribution, function, and evolution of macromolecular systems. As the number of available genomes continues to grow, tools like PANORAMA will become increasingly critical for unveiling the complex architectures of microbial defense and other macromolecular systems.

Looking ahead, we plan to expand PANORAMA's detection capabilities by incorporating anti-defense system models [36] from DefenseFinder (v1.3.0), enabling analysis of the co-evolutionary dynamics between defense and anti-defense mechanisms. To identify genomic contexts that encode enzymes involved in the same metabolic pathway, we plan to incorporate additional rules based on comprehensive databases such as KEGG [37] or MetaCyc [38]. In parallel, we will develop dedicated rules to detect clusters of Carbohydrate-Active Enzymes (CAZymes) involved in polysaccharide biosynthesis and degradation.

All these model collections will be made publicly available on GitHub (https://github.com/PANORAMA-models) and, following the approach of other tools in the field, we will implement functionality within PANORAMA to automatically download the latest version of these models, simplifying the functional analysis of pangenomes. Furthermore, the pangenome graph framework provided by PANORAMA offers a promising avenue to explore genomic context and uncover novel systems. By incorporating fuzzy functional predictors based on structural similarity or protein language models, we hope to detect new systems composed of functionally analogous components co-localized within the pangenome graph.

Materials and methods

Overview of PANORAMA software

PANORAMA is an open-source Python 3 software (version ≥3.10 required) that enables pangenome-scale detection and analysis of prokaryotic macromolecular systems and pangenome comparison to identify similar structures based on gene family content. The tool is freely available on GitHub (https://github.com/labgem/PANORAMA) and is distributed under the CeCILL 2.1 license. Built on the PPanGGOLiN API version 2.3.0, PANORAMA inherits its core concepts and definitions for pangenome partitions, gene families, RGPs, and spots of insertion [8,9]. The software can be easily installed via bioconda [39] (https://anaconda.org/bioconda/panorama), ensuring reproducible deployments across different computing

environments. Comprehensive documentation is available online at https://panorama.readthedocs.io/latest/, including detailed analysis workflows, parameter descriptions, usage examples, and a step-by-step guide for creating custom detection models.

Pangenome system detection workflow

To predict macromolecular systems, PANORAMA employs rule-based models that analyze the presence/absence of specific functions predicted from pangenome gene families while considering constraints on their genomic organization. The PanSystem workflow begins with gene family annotation using HMM libraries, followed by the identification and evaluation of genomic contexts within the pangenome graph based on system model rules (Fig 9). Finally, the predicted systems at the pangenome level are mapped onto individual genomes to assess their presence and gene content.

System modeling. Models used by PANORAMA for predicting macromolecular systems follow the same grammar and logic as those of MacSyFinder [15], but introduce an additional hierarchical level and extended parameters to describe system architecture with greater biological accuracy (Fig 10). The primary components of a system *Model* are *Families* (*i.e.*, isofunctional protein families) instead of genes, and an additional hierarchical level is introduced to represent *Functional Units*. A *Functional Unit* is defined as a set of *Families* that work together to perform a necessary function for the system. Several *Functional Units* may be required for a system to operate effectively. This new level provides a more detailed and accurate description of systems, allowing the use of distinct rules for presence/absence and genomic organization, both for the *Families* of a *Functional Unit* and between *Functional Units*. In PANORAMA, we also introduce the concept of canonical *Models*, which represent a more relaxed definition of systems. In PADLOC [16], these models are identified by the keyword "other" in their names. While these systems may not have been experimentally validated and might not be functional, their predictions can be valuable for identifying new systems or potential variants. During system detection, priority is given to non-canonical models. A canonical model is predicted only if its *Families* are not already associated with a non-canonical model.

For presence/absence constraints, each *Functional Unit* and *Family* is categorized as *mandatory*, *accessory*, *neutral*, or *forbidden*. *Mandatory* elements are essential for the system. *Accessory* components are dispensable. They contribute to the system, but may not be identified in all system variants due to rapid evolution or the absence of homology. *Forbidden* elements are incompatible with system functionality. They can help differentiate systems with shared components or distinguish inhibited systems. Although *Neutral* components are considered associated functions, they are not used to assess system predictions. However, they can serve as intermediaries in genomic context analysis by linking mandatory

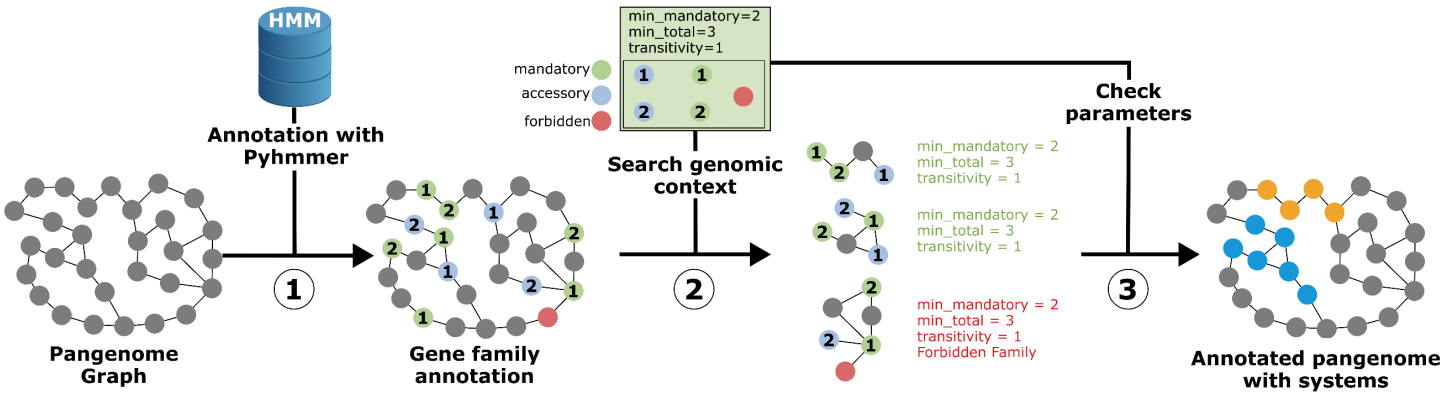

Fig 9. PANORAMA PanSystem workflow. The detection workflow for macromolecular systems consists of multiple sequential steps: (1) gene family annotation using HMM profiles, (2) detection of genomic contexts from the pangenome graph containing annotated families, (3) checking model rules.

https://doi.org/10.1371/journal.pcbi.1013856.g009

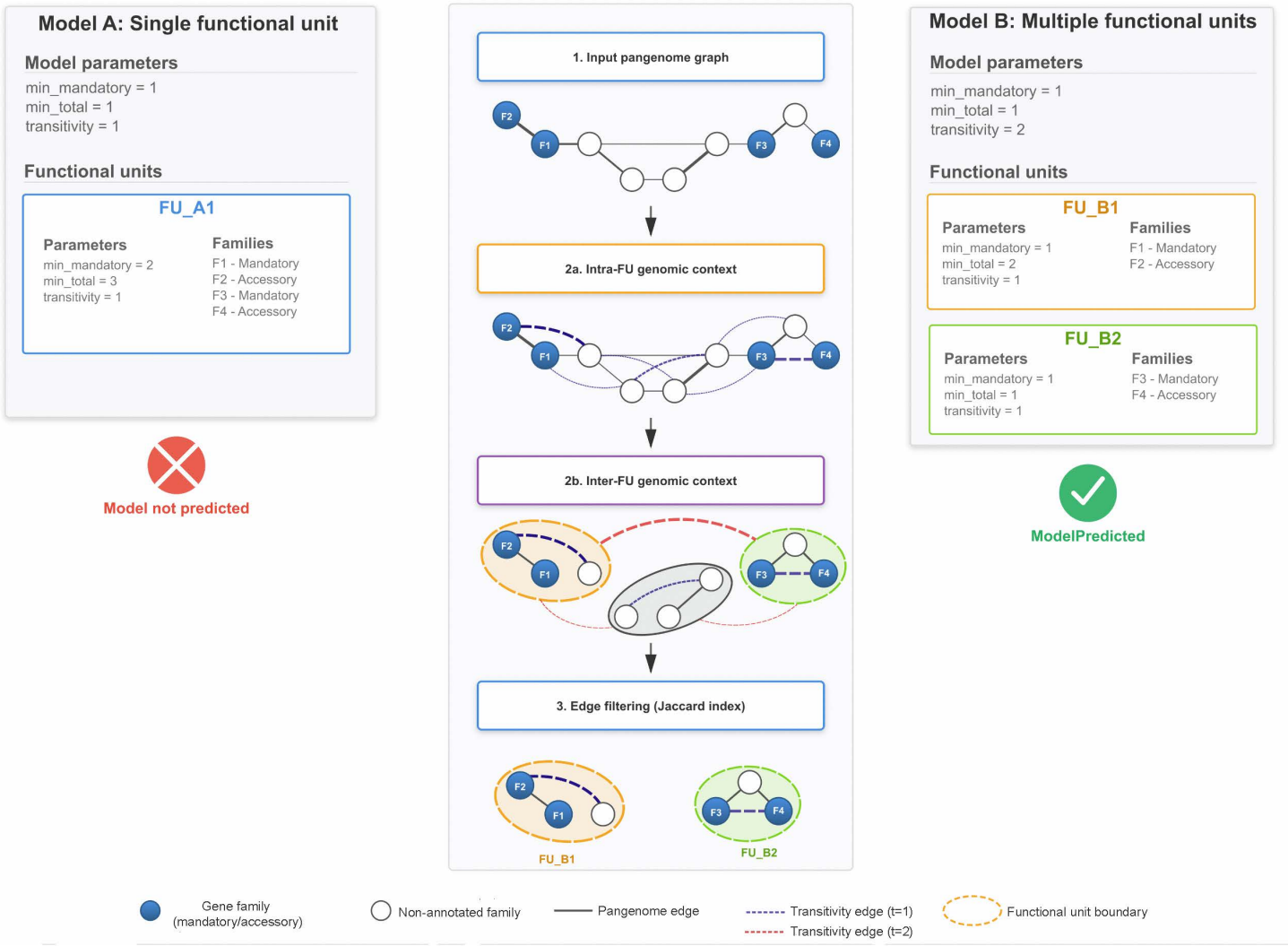

Fig 10. PANORAMA system modeling. Two modeling approaches are compared: Model A (left) uses a single functional unit with parameters that fail to detect the structure (*min_ mandatory* = 1, *min_ total* = 1, *transitivity* = 1), while Model B (right) employs multiple functional units with increased transitivity (*transitivity* = 2), successfully identifying FU_ B1 (orange) and FU_ B2 (green). Blue nodes represent gene families within the model (mandatory or accessory), white nodes indicate gene family without function for the system, solid black lines show pangenome edges, and dashed lines represent transitivity edges connecting genes at specified graph distances. Colored dashed ovals delineate predicted functional unit boundaries.

https://doi.org/10.1371/journal.pcbi.1013856.g010

or accessory elements. To predict a system, quorum rules on component presence/absence are defined by two parameters: *minimum_ mandatory* (the minimum number of mandatory elements required) and *minimum_ total* (the minimum number of both mandatory and accessory elements). These parameters are specified at two levels: at the *Model* level, to assess the presence of *Functional Units*, and at the *Functional Unit* level, to evaluate predicted *Families*. Constraints on the genomic organization of a system are governed by a *transitivity* parameter, which defines the maximum genomic distance between gene families in the pangenome graph corresponding to the system components (see Pangenome context extraction).

Model creation and distribution. PANORAMA provides a comprehensive set of utilities to support model creation, validation, and distribution. For users who wish to reuse existing model collections, an automatic translation command

(https://panorama.readthedocs.io/latest/modeler/translate.html) converts models from MacSyFinder [15], DefenseFinder [17], CasFinder [32], and PADLOC [16] into the PANORAMA JSON format, together with the associated HMM database input files, without requiring manual reformatting. For users building new models from scratch, dedicated utilities generate both the model and the corresponding HMM database input files from user-supplied data (https://panorama.readthedocs.io/latest/modeler/modeling.html). In both cases, PANORAMA automatically validates the grammar and internal consistency of all model files before execution, reporting errors and warnings so that any issues can be corrected prior to analysis.

Ready-to-use model collections and their associated HMM libraries are publicly available (https://github.com/PANORAMA-models), allowing users to download and apply curated detection models directly without additional configuration. This repository follows an open contribution model: researchers who develop new system models are encouraged to submit them for distribution, enabling the community to continuously expand the detection capabilities of PANORAMA.

Functional annotation of pangenome gene families. To annotate the pangenome graph, PANORAMA utilizes HMM libraries, in which each HMM represents a specific *Family* defined in the system model. By default, the protein sequences of each family's representative genes are aligned to the HMM profiles using the hmmsearch method from the pyHMMER Python library [33]. Since gene families in the pangenome are constructed with default thresholds of 80% amino acid identity and 80% sequence coverage, members within a family share a high degree of sequence similarity, which limits the risk that the representative sequence fails to capture the functional breadth of the family. Nevertheless, PANORAMA provides two additional alignment strategies to address cases where intra-family sequence diversity may be relevant. First, a consensus protein sequence of each gene family can be used as the alignment target; it is computed by performing a multiple sequence alignment (MSA) internally with MAFFT [40], excluding fragmented genes, or by reading an externally provided MSA and then computing the consensus sequence with pyHMMER. Second, HMM profiles can be aligned against all individual gene sequences within each family, ensuring that no divergent member is overlooked, at the cost of increased computation time. To validate alignments, a metadata file can be linked to an HMM library, specifying thresholds for each HMM based on various criteria, such as alignment coverage on the sequence or profile, score, e-value, and independent e-value. Alternatively, a global threshold can be applied to all HMMs. An option is also available to keep only the *n* best hits per family. Gene family annotation can also be performed externally using alternative prediction methods, with results provided to PANORAMA as a tab-separated values (TSV) file. Multiple annotation sources can be used simultaneously, enabling combined analyses — for example, applying a standard database such as DefenseFinder models alongside a user-defined custom HMM library in a single run. This is particularly relevant for functions not yet represented in existing HMM databases, for which external annotation tools or purpose-built HMM profiles can be used upstream of PANORAMA and supplied via this mechanism.

Pangenome context extraction. For each system, connected components between gene families corresponding to model families (hereafter called target families) are searched for in the pangenome graph. According to the model rules on genomic colocalization of system components (*i.e.*, the *transitivity* parameter), additional edges are added to the pangenome graph between families separated by fewer than *t* genes in the corresponding genomes. This corresponds to a partial transitive closure on the pangenome graph, enabling the connection of two target families even if their genes are not directly adjacent in the genomes. Next, a Jaccard index-based criterion (Eq 1) is computed on edges to evaluate genomic context conservation (*i.e.*, synteny conservation). For two families *a* and *b* connected by an edge $e_{a,b}$, the index $J_{a,b}$ is defined as the ratio of the number of genomes in which the edge $e_{a,b}$ exists to the number of genomes in which at least one of the two gene families is present ($|\text{gen}_a \cup \text{gen}_b|$). To enhance the detection of systems present in a limited number of genomes, this index is computed locally, considering only the genomes that contain the target families of the connected component rather than all the genomes of the pangenome. Edges having a Jaccard index below a defined threshold ($J_{a,b} < 0.8$ by default) are removed. The connected components with their remaining nodes are then evaluated

in terms of presence/absence rules to validate system predictions. For each connected component, this process, which combines edge filtering and presence/absence rule validation, is applied iteratively, starting with the largest set of target families observed in a genome and then exploring other sets of target families, from largest to smallest, if they are not already included in a predicted system.

$$J_{a,b} = \frac{|\text{gen}_{e_{a,b}}|}{|\text{gen}_a \cup \text{gen}_b|} \quad (1)$$

Finally, the connected components corresponding to predicted systems at the pangenome level are mapped back onto individual genomes. Their occurrence in individual strains is assessed based on presence/absence and colocalization rules. They are classified into three distinct categories: *strict*, *extended*, or *split*. A system is flagged as *strict* if the *transitivity* parameter is respected among the genes belonging to the system. If additional genes of the connected component are found among those encoding the system, it is classified as *extended*. Indeed, applying transitive closure to the pangenome graph enables the detection of additional genes conserved with those of the system. This provides a more relaxed definition of colocalization rules than the strict intergenic space constraints employed by MacSyFinder or PADLOC. Lastly, *split* systems are those in which system genes are separated by non-member genes of the connected component detected at the pangenome level or are located on different contigs. Such fragmentation can occur in a subset of genomes due to rearrangements, insertion events (*e.g.*, insertion sequences), or assembly breaks.

Pangenome comparison workflow

The PanCompare workflow of PANORAMA enables the comparison of pangenomes and the identification of similar elements across species, such as macromolecular systems or spots of insertion. These elements are represented as sets of gene families. The first step is to cluster all the gene families from the different pangenomes to identify groups of homologous gene families (GHGFs). Then, a Gene Family Repertoire Relatedness score (GFRR) for each pair of elements is computed. Finally, a community clustering algorithm is applied to identify clusters of elements sharing similar gene family content.

Gene families clustering. The process involves the extraction of representative protein sequences of gene families from all pangenomes. These sequences are then clustered using the MMSeqs2 cluster command [41] to obtain GHGFs sharing at least 50% of amino acid identity (*–min-seq-id* parameter) with an alignment coverage of 80% (*-c* parameter) by default. These thresholds are consistent with those used in Uniclust50 [42], a widely used protein sequence clustering database that demonstrates high functional annotation consistency at 50% identity. The following additional parameters are used: *–max-seqs* 400 *–min-ungapped-score* 1 *–kmer-per-seq* 80 *–alignment-mode* 2 *–cluster-mode* 1.

Gene family repertoire relatedness score. To compare the family content of two pangenome elements, two GFRR scores are computed using GHGFs. For two pangenome elements, *a* and *b*, with their GHGFs content denoted as $GHGF_a$ and $GHGF_b$, the minimal GFRR score, $minGFRR_{a,b}$, is defined as the ratio of the number of gene families belonging to the same GHGFs in *a* and *b* to the minimum number of gene families between *a* and *b* (Eq 2). Similarly, the maximal GFRR score, $maxGFRR_{a,b}$, is computed using the maximum number of gene families between *a* and *b* (Eq 3).

$$minGFRR_{a,b} = \frac{|\text{GHGF}_a \cap \text{GHGF}_b|}{\min(|\text{GHGF}_a|, |\text{GHGF}_b|)} \quad (2)$$

$$maxGFRR_{a,b} = \frac{|\text{GHGF}_a \cap \text{GHGF}_b|}{\max(|\text{GHGF}_a|, |\text{GHGF}_b|)} \quad (3)$$

GFRR score computation can be applied to predicted macromolecular systems or spots of insertion. For spots, the two sets of gene families corresponding to spot borders are merged to compute the GFRR scores.

Pangenome element clustering. To identify similar elements (*i.e.*, spots of insertion or systems) between pangenomes, GFRR scores are computed for each pair of elements. A graph is then constructed, where nodes represent pangenome elements and edges are weighted by their corresponding GFRR scores. Edges are filtered according to a GFRR threshold (min_{GFRR} >= 0.6 by default) to ensure strong similarity between connected elements. Finally, a Louvain algorithm [43], using the NetworkX library [44] implementation with GFRR weights on edges, is applied to the graph to identify non-overlapping communities corresponding to groups of similar elements between pangenomes. This functionality was used to identify conserved spots of insertion between species containing defense systems. A system is associated with a spot if all of its gene families are part of RGPs found within the boundaries of that spot.

Data, benchmark, and metrics

Genomic data and defense system prediction. Complete genomes of five species were downloaded from NCBI RefSeq [45] and analyzed for defense system prediction using DefenseFinder (v1.2.2 with 239 models of v1.2.4 + 43 CasFinder models of v3.1.0), PADLOC (v2.0.0, 385 models of v2.0.0), and PANORAMA (v1.0.0). The dataset includes *Pseudomonas aeruginosa* (941 genomes) and four well-studied Enterobacteriaceae species: *Citrobacter freundii* (79 genomes), *Escherichia coli* (3,083 genomes), *Klebsiella pneumoniae* (1,659 genomes), and *Salmonella enterica* (1,380 genomes). Before running PANORAMA, the pangenomes of the five species were obtained using PPanGGOLiN v2.3.0 with default parameters while keeping the original RefSeq annotations. Pangenome metrics, including the number of families categorized as *persistent*, *shell*, or *cloud* genome, as well as the number of predicted RGPs and spots of insertion, are summarized in Table 3. These metrics are also available through the info command of PANORAMA. Models from DefenseFinder and PADLOC (n = 667), along with their associated HMMs (n = 6,272), were converted to PANORAMA format using its internal conversion utility.

Benchmark protocol

The *P. aeruginosa* genome dataset was used to evaluate PANORAMA's defense system predictions against the reference tools, DefenseFinder v1.2.2 and PADLOC v2.0.0, using their respective system models and HMMs. To ensure consistency in the comparison, the predictions from DefenseFinder and PADLOC, originally consisting of gene sets associated with defense systems, were mapped to the pangenome gene families by linking each identified gene to its corresponding family. Then, we defined a True Positive (TP) as a gene family assigned to the same defense system by both PANORAMA and the reference tool. Gene families associated with a system by the reference tool but not predicted by PANORAMA are classified as False Negatives (FN). Gene families assigned to a different system or not predicted by the reference tool are considered False Positives (FP). To assess the performance of PANORAMA, we computed precision, recall, and F1-score using the following equations:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{4a}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4b}$$

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4c}$$

The benchmark was conducted on a dedicated Linux server equipped with two Intel Xeon Gold 6150 processors (36 cores), 376 GiB of DDR4 RAM, running CentOS v7.9.2009 with kernel 3.10.0 and Python 3.10. To ensure a fair comparison and consistent resource availability, the entire machine was reserved for each tool execution. DefenseFinder and PADLOC were run sequentially, processing one genome at a time on a single CPU core, with exclusive access to all available memory. Similarly, PANORAMA was run with the *–threads* parameter set to 1 with dedicated machine access. To evaluate execution performance, total run time and peak memory usage were measured for each tool. For completeness, the pangenome construction step performed by PPanGGOLiN is reported separately in S1 Table as a prerequisite step for PANORAMA. Although raw genomes are indeed the starting point for all tools, we note that the PPanGGOLiN pangenome is built once per species and its output can be reused across any number of subsequent pangenome analyses, amortizing its construction cost.

System category diversity. To assess the compositional diversity of systems within a given category (*e.g.*, Restriction Modification, Gabija, CRISPR-Cas) in a pangenome, Shannon entropy is calculated as follows:

$$H(C,p) = -\sum_{i \in C} P(sm_i) \log_2 P(sm_i) \tag{5}$$

where:

- $H(C,p)$: the Shannon entropy of a system category C in a pangenome p.
- $P(sm_i)$: probability of a system model sm_i of C occurring in p, calculated for each system model as the ratio of the number of gene families of p associated with sm_i to the total number of gene families across all systems of C.

Higher entropy indicates greater diversification of the gene families comprising the systems within a given category.

System category enrichment. To determine the specificity of a system category in a set of pangenomes, an enrichment factor is computed as follows:

$$EF(C,p) = \frac{f_r(C,p)}{f_r(C,P)} \tag{6}$$

where:

- C: a system category.
- P: a collection of pangenomes p.
- $f_r(C,p)$: the relative frequency of the system category C in the pangenome p, calculated as the ratio of the number of systems of C predicted in p to the total number of systems in p.
- $f_r(C,P)$: the relative frequency of the system category C in P, calculated as the ratio of the number of systems of C predicted in all $p \in P$ to the total number of systems across all $p \in P$.

An enrichment factor above 1 indicates that a system category is more present in a specific pangenome than in the entire collection of pangenomes.

Data visualization

PANORAMA interactive outputs are generated using the Bokeh library and can be explored directly in a web browser. All static figures presented in this manuscript were generated by the authors using matplotlib (v3.4) [46] and seaborn (v0.13.2) [47]. UpSet plots were produced using the upsetplot library (v0.9.0 - https://github.com/jnothman/UpSetPlot/)

[48]. Color palettes were selected to be colorblind-friendly using seaborn and colorcet (v3.1.0 - https://github.com/holoviz/colorcet). Data analysis was performed using numpy (v2.4.0) [49], pandas (v2.2.3) [50], and networkx (v3.4.2) [44]. All figure generation scripts and data analyses are compiled in a Jupyter Notebook (jupyter v1.1.1, ipykernel v6.29.5) [51]. Conceptual diagrams (Figs 1, 9, and 10) were created by the authors using inkscape (https://inkscape.org) or draw.io (https://www.drawio.com), two open-source diagramming tools released under the Apache License 2.0.

Supporting information

S1 Fig. Computational performance of the analysis pipeline across bacterial species. Stacked bar chart showing the execution time (in minutes) for each step of the workflow across species with varying numbers of genomes (black line, right y-axis). The four pipeline steps are: Load (blue) - time to load the pangenome data; Annotation (green) - time to annotate gene families with defense system function with HMMs; Detection (yellow) - time to detect defense systems; and Projection (red) - time to project defense systems across genomes. The benchmark was conducted on a Linux server with two Intel Xeon Gold 6150 processors (36 cores), 376 GiB RAM, running CentOS v7.9.2009 with Python 3.10. Execution time scales with dataset size, with the projection step exhibiting super-linear scaling.
(PDF)

S1 Table. Execution time of the workflow across multiple species. Execution time (in minutes:seconds format) for each step of the analysis workflow across five bacterial species with varying dataset sizes. PPanGGOLiN: time to construct, partition and analyse (RGPs, Spots and modules) the pangenome graph; Load: time to load the pangenome data; Annotation: time to annotate gene families with defense system functions from HMMs; Detection: time to detect defense systems; Projection: time to project defense systems across genomes. Analyses were performed on two Intel Xeon Gold 6150 processors (36 cores), 376 GiB RAM, with Python 3.10. Execution time scales with the number of genomes, with the projection step showing the most substantial increase in larger datasets. *P. aeruginosa* exception can be explained by the use of fasta file in the pangenome construction, adding an annotation step in PPanGGOLiN workflow.
(PDF)

S2 Fig. Example of PANORAMA automated output: number of systems predicted per model across Enterobacteriaceae pangenomes. This heatmap is automatically generated by PANORAMA using the Bokeh package [53]. Each row corresponds to a species pangenome and each column to a defense system model; color intensity indicates the number of predicted systems (color scale, right). The figure is shown here to illustrate the format of PANORAMA's built-in visualization output. The full interactive version — enabling zooming, panning, and hovering to display exact values — is available in the Zenodo repository and can be opened directly in any standard web browser.
(PDF)

S3 Fig. Species-specificity evaluation across system categories in Enterobacteriaceae pangenomes. The enrichment factors were computed using the method described in Eq 6, providing a quantitative measure of species-specific representation.
(PDF)

S4 Fig. Defense systems within insertion spots. The bar plot displays the number of predicted defense systems in the pangenome for each spot of insertion. (A) *E. coli* pangenome. (B) *S. enterica* pangenome.
(PDF)

S5 Fig. Defense system composition of spots belonging to cluster 3062. Distribution of defense systems in two representative spots from cluster 3062: spot 175 in *E. coli* (A) and spot 86 in *S. enterica* (B). Each slice represents the proportion of a given defense system category within the spot. RM (Restriction-Modification) systems dominate both

spots, accounting for 81.3% in *E. coli* and 57.1% in *S. enterica*. BREX represents the second most abundant system in *S. enterica* (23.8%), while it is present at lower frequency in *E. coli* (6.6%). Other defense systems are present at lower frequencies.
(PDF)

Acknowledgments

We are grateful to Nicolas Wiart for technical support and assistance in testing the software on the computing infrastructure, to Sophie Abby for her help and advice on system modeling, and to Violette Da Cunha for insightful discussions that helped refine the methodological choices during tool development.

Author contributions

Conceptualization: Jérôme Arnoux, David Vallenet, Alexandra Calteau.

Data curation: Jérôme Arnoux.

Formal analysis: Jérôme Arnoux.

Funding acquisition: David Vallenet, Alexandra Calteau.

Investigation: Jérôme Arnoux, Laura Bry, Quentin Fernandez de Grado, Yazid Hoblos.

Methodology: Jérôme Arnoux, Jean Mainguy, Laura Bry, Quentin Fernandez de Grado, Yazid Hoblos.

Project administration: David Vallenet, Alexandra Calteau.

Software: Jérôme Arnoux, Jean Mainguy, Laura Bry, Quentin Fernandez de Grado, Yazid Hoblos.

Supervision: David Vallenet, Alexandra Calteau.

Validation: Jérôme Arnoux, Quentin Fernandez de Grado, Yazid Hoblos.

Visualization: Jérôme Arnoux.

Writing – original draft: Jérôme Arnoux.

Writing – review & editing: Jérôme Arnoux, Jean Mainguy, Laura Bry, Quentin Fernandez de Grado, Yazid Hoblos, David Vallenet, Alexandra Calteau.

References

1. Land M, Hauser L, Jun S-R, Nookaew I, Leuze MR, Ahn T-H, et al. Insights from 20 years of bacterial genome sequencing. Funct Integr Genomics. 2015;15(2):141–61. https://doi.org/10.1007/s10142-015-0433-4 PMID: 25722247
2. Parks DH, Chuvochina M, Waite DW, Rinke C, Skarshewski A, Chaumeil P-A, et al. A standardized bacterial taxonomy based on genome phylogeny substantially revises the tree of life. Nat Biotechnol. 2018;36(10):996–1004. https://doi.org/10.1038/nbt.4229 PMID: 30148503
3. Tettelin H, Masignani V, Cieslewicz MJ, Donati C, Medini D, Ward NL, et al. Genome analysis of multiple pathogenic isolates of Streptococcus agalactiae: implications for the microbial "pan-genome". Proc Natl Acad Sci U S A. 2005;102(39):13950–5. https://doi.org/10.1073/pnas.0506758102 PMID: 16172379
4. Vernikos G, Medini D, Riley DR, Tettelin H. Ten years of pan-genome analyses. Curr Opin Microbiol. 2015;23:148–54. https://doi.org/10.1016/j.mib.2014.11.016 PMID: 25483351
5. The Computational Pan-Genomics Consortium. Computational pan-genomics: status, promises and challenges. Brief Bioinform. 2018;19(1):118–35. https://doi.org/10.1093/bib/bbw089
6. Holley G, Melsted P. Bifrost: highly parallel construction and indexing of colored and compacted de Bruijn graphs. Genome Biol. 2020;21(1):249. https://doi.org/10.1186/s13059-020-02135-8 PMID: 32943081
7. Garrison E, Sirén J, Novak AM, Hickey G, Eizenga JM, Dawson ET, et al. Variation graph toolkit improves read mapping by representing genetic variation in the reference. Nat Biotechnol. 2018;36(9):875–9. https://doi.org/10.1038/nbt.4227 PMID: 30125266
8. Gautreau G, Bazin A, Gachet M, Planel R, Burlot L, Dubois M, et al. PPanGGOLiN: Depicting microbial diversity via a partitioned pangenome graph. PLoS Comput Biol. 2020;16(3):e1007732. https://doi.org/10.1371/journal.pcbi.1007732 PMID: 32191703

9. Bazin A, Gautreau G, Médigue C, Vallenet D, Calteau A. panRGP: a pangenome-based method to predict genomic islands and explore their diversity. Bioinformatics. 2020;36(Suppl_2):i651–8. https://doi.org/10.1093/bioinformatics/btaa792 PMID: 33381850
10. Bazin A, Medigue C, Vallenet D, Calteau A. panModule: detecting conserved modules in the variable regions of a pangenome graph. bioRxiv. 2021. https://doi.org/10.1101/2021.12.06.471380
11. Whelan FJ, Rusilowicz M, McInerney JO. Coinfinder: detecting significant associations and dissociations in pangenomes. Microb Genom. 2020;6(3):e000338. https://doi.org/10.1099/mgen.0.000338 PMID: 32100706
12. Gavriilidou A, Paulitz E, Resl C, Ziemert N, Kupczok A, Baumdicker F. Goldfinder: Unraveling Networks of Gene Co-occurrence and Avoidance in Bacterial Pangenomes. bioRxiv. 2024. 2024.04.29.591652 p. https://www.biorxiv.org/content/10.1101/2024.04.29.591652v1
13. Mariano G, Trunk K, Williams DJ, Monlezun L, Strahl H, Pitt SJ, et al. A family of Type VI secretion system effector proteins that form ion-selective pores. Nat Commun. 2019;10(1):5484. https://doi.org/10.1038/s41467-019-13439-0 PMID: 31792213
14. Zhang R, Mirdita M, Söding J. De novo discovery of conserved gene clusters in microbial genomes with Spacedust. Nat Methods. 2025;22(10):2065–73. https://doi.org/10.1038/s41592-025-02816-x PMID: 40954296
15. Néron B, Denise R, Coluzzi C, Touchon M, Rocha EPC, Abby SS. MacSyFinder v2: Improved modelling and search engine to identify molecular systems in genomes. Peer Commun J. 2023;3. https://doi.org/10.24072/pcjournal.250
16. Payne LJ, Todeschini TC, Wu Y, Perry BJ, Ronson CW, Fineran PC, et al. Identification and classification of antiviral defence systems in bacteria and archaea with PADLOC reveals new system types. Nucleic Acids Res. 2021;49(19):10868–78. https://doi.org/10.1093/nar/gkab883 PMID: 34606606
17. Tesson F, Hervé A, Mordret E, Touchon M, d'Humières C, Cury J, et al. Systematic and quantitative view of the antiviral arsenal of prokaryotes. Nat Commun. 2022;13(1):2561. https://doi.org/10.1038/s41467-022-30269-9 PMID: 35538097
18. Tonkin-Hill G, MacAlasdair N, Ruis C, Weimann A, Horesh G, Lees JA, et al. Producing polished prokaryotic pangenomes with the Panaroo pipeline. Genome Biol. 2020;21(1):180. https://doi.org/10.1186/s13059-020-02090-4 PMID: 32698896
19. Jonkheer EM, van Workum D-JM, Sheikhizadeh Anari S, Brankovics B, de Haan JR, Berke L, et al. PanTools v3: functional annotation, classification and phylogenomics. Bioinformatics. 2022;38(18):4403–5. https://doi.org/10.1093/bioinformatics/btac506 PMID: 35861394
20. Bertani G, Weigle JJ. Host controlled variation in bacterial viruses. J Bacteriol. 1953;65(2):113–21. https://doi.org/10.1128/jb.65.2.113-121.1953 PMID: 13034700
21. Makarova KS, Wolf YI, Iranzo J, Shmakov SA, Alkhnbashi OS, Brouns SJJ, et al. Evolutionary classification of CRISPR-Cas systems: a burst of class 2 and derived variants. Nat Rev Microbiol. 2020;18(2):67–83. https://doi.org/10.1038/s41579-019-0299-x PMID: 31857715
22. Goldfarb T, Sberro H, Weinstock E, Cohen O, Doron S, Charpak-Amikam Y, et al. BREX is a novel phage resistance system widespread in microbial genomes. EMBO J. 2015;34(2):169–83. https://doi.org/10.15252/embj.201489455 PMID: 25452498
23. Ofir G, Melamed S, Sberro H, Mukamel Z, Silverman S, Yaakov G, et al. DISARM is a widespread bacterial defence system with broad anti-phage activities. Nat Microbiol. 2018;3(1):90–8. https://doi.org/10.1038/s41564-017-0051-0 PMID: 29085076
24. Doron S, Melamed S, Ofir G, Leavitt A, Lopatina A, Keren M, et al. Systematic discovery of antiphage defense systems in the microbial pangenome. Science. 2018;359(6379):eaar4120. https://doi.org/10.1126/science.aar4120 PMID: 29371424
25. Millman A, Bernheim A, Stokar-Avihail A, Fedorenko T, Voichek M, Leavitt A. Bacterial retrons function in anti-phage defense. Cell. 2020;183(6):1551–61.e12. https://doi.org/10.1016/j.cell.2020.09.065
26. Georjon H, Bernheim A. The highly diverse antiphage defence systems of bacteria. Nat Rev Microbiol. 2023;21(10):686–700. https://doi.org/10.1038/s41579-023-00934-x PMID: 37460672
27. Koonin EV, Makarova KS, Wolf YI. Evolutionary Genomics of Defense Systems in Archaea and Bacteria. Annu Rev Microbiol. 2017;71:233–61. https://doi.org/10.1146/annurev-micro-090816-093830 PMID: 28657885
28. Hochhauser D, Millman A, Sorek R. The defense island repertoire of the Escherichia coli pan-genome. PLoS Genet. 2023;19(4):e1010694. https://doi.org/10.1371/journal.pgen.1010694 PMID: 37023146
29. Eddy SR. Accelerated Profile HMM Searches. PLoS Comput Biol. 2011;7(10):e1002195. https://doi.org/10.1371/journal.pcbi.1002195 PMID: 22039361
30. Bastian M, Heymann S, Jacomy M. Gephi: An Open Source Software for Exploring and Manipulating Networks. ICWSM. 2009;3(1):361–2. https://doi.org/10.1609/icwsm.v3i1.13937
31. Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, et al. Cytoscape: a software environment for integrated models of biomolecular interaction networks. Genome Res. 2003;13(11):2498–504. https://doi.org/10.1101/gr.1239303 PMID: 14597658
32. Couvin D, Bernheim A, Toffano-Nioche C, Touchon M, Michalik J, Néron B, et al. CRISPRCasFinder, an update of CRISRFinder, includes a portable version, enhanced performance and integrates search for Cas proteins. Nucleic Acids Res. 2018;46(W1):W246–51. https://doi.org/10.1093/nar/gky425 PMID: 29790974
33. Larralde M, Zeller G. PyHMMER: a Python library binding to HMMER for efficient sequence analysis. Bioinformatics. 2023;39(5):btad214. https://doi.org/10.1093/bioinformatics/btad214 PMID: 37074928

34. Johnson MC, Laderman E, Huiting E, Zhang C, Davidson A, Bondy-Denomy J. Core defense hotspots within Pseudomonas aeruginosa are a consistent and rich source of anti-phage defense systems. Nucleic Acids Res. 2023;51(10):4995–5005. https://doi.org/10.1093/nar/gkad317 PMID: 37140042
35. Lescat M, Calteau A, Hoede C, Barbe V, Touchon M, Rocha E, et al. A module located at a chromosomal integration hot spot is responsible for the multidrug resistance of a reference strain from Escherichia coli clonal group A. Antimicrob Agents Chemother. 2009;53(6):2283–8. https://doi.org/10.1128/AAC.00123-09 PMID: 19364861
36. Tesson F, Huiting E, Wei L, Ren J, Johnson M, Planel R, et al. Exploring the diversity of anti-defense systems across prokaryotes, phages and mobile genetic elements. Nucleic Acids Res. 2025;53(1):gkae1171. https://doi.org/10.1093/nar/gkae1171 PMID: 39657785
37. Kanehisa M, Furumichi M, Sato Y, Matsuura Y, Ishiguro-Watanabe M. KEGG: biological systems database as a model of the real world. Nucleic Acids Res. 2025;53(D1):D672–7. https://doi.org/10.1093/nar/gkae909 PMID: 39417505
38. Caspi R, Billington R, Keseler IM, Kothari A, Krummenacker M, Midford PE, et al. The MetaCyc database of metabolic pathways and enzymes - a 2019 update. Nucleic Acids Res. 2020;48(D1):D445–53. https://doi.org/10.1093/nar/gkz862 PMID: 31586394
39. Grüning B, Dale R, Sjödin A, Chapman BA, Rowe J, Tomkins-Tinch CH, et al. Bioconda: sustainable and comprehensive software distribution for the life sciences. Nat Methods. 2018;15(7):475–6. https://doi.org/10.1038/s41592-018-0046-7 PMID: 29967506
40. Katoh K, Standley DM. MAFFT multiple sequence alignment software version 7: improvements in performance and usability. Mol Biol Evol. 2013;30(4):772–80. https://doi.org/10.1093/molbev/mst010 PMID: 23329690
41. Steinegger M, Söding J. Clustering huge protein sequence sets in linear time. Nat Commun. 2018;9(1):2542. https://doi.org/10.1038/s41467-018-04964-5 PMID: 29959318
42. Mirdita M, von den Driesch L, Galiez C, Martin MJ, Söding J, Steinegger M. Uniclust databases of clustered and deeply annotated protein sequences and alignments. Nucleic Acids Res. 2017;45(D1):D170–6. https://doi.org/10.1093/nar/gkw1081 PMID: 27899574
43. Blondel VD, Guillaume J-L, Lambiotte R, Lefebvre E. Fast unfolding of communities in large networks. J Stat Mech. 2008;2008(10):P10008. https://doi.org/10.1088/1742-5468/2008/10/p10008
44. Hagberg AA, Schult DA, Swart PJ. Exploring Network Structure, Dynamics, and Function using NetworkX. In: Varoquaux G, Vaught T, Millman J, editors. Proceedings of the 7th Python in Science Conference. Pasadena, CA USA. 2008. p. 11–15.
45. O'Leary NA, Wright MW, Brister JR, Ciufo S, Haddad D, McVeigh R, et al. Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. Nucleic Acids Res. 2016;44(D1):D733-45. https://doi.org/10.1093/nar/gkv1189 PMID: 26553804
46. Hunter JD. Matplotlib: A 2D Graphics Environment. Comput Sci Eng. 2007;9(3):90–5. https://doi.org/10.1109/MCSE.2007.55
47. Waskom M. seaborn: statistical data visualization. JOSS. 2021;6(60):3021. https://doi.org/10.21105/joss.03021
48. Lex A, Gehlenborg N, Strobelt H, Vuillemot R, Pfister H. UpSet: Visualization of Intersecting Sets. IEEE Trans Vis Comput Graph. 2014;20(12):1983–92. https://doi.org/10.1109/TVCG.2014.2346248 PMID: 26356912
49. Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, et al. Array programming with NumPy. Nature. 2020;585(7825):357–62. https://doi.org/10.1038/s41586-020-2649-2 PMID: 32939066
50. team Tpd. pandas-dev/pandas: Pandas. Zenodo; 2025. Available from: https://zenodo.org/records/17229934
51. Granger BE, Perez F. Jupyter: Thinking and Storytelling With Code and Data. Comput Sci Eng. 2021;23(2):7–14. https://doi.org/10.1109/mcse.2021.3059263
52. European Organization For Nuclear Research, OpenAIRE. Zenodo: Research. Shared. CERN; 2013. Available from: https://www.zenodo.org/
53. Bokeh Development Team. Bokeh: Python library for interactive visualization; 2018. Available from: https://bokeh.pydata.org/en/latest/