

Top MCP Servers for Biotech: Connecting AI to Research Data

By Adrien Laurent, CEO at IntuitionLabs • 10/27/2025 • 45 min read

model context protocol

mcp servers

biotechnology

ai agents

large language models

drug discovery

bioinformatics

api standards

genomics



Top MCP Servers for Biotech: Connecting AI to Research Data

intuitionlabs.ai

Executive Summary

The **Model Context Protocol (MCP)** is an emerging open standard (endorsed by Anthropic, OpenAI, Google, Microsoft, and others) designed to let AI systems (especially large language models, LLMs) seamlessly access and interact with external data sources and tools. In biotechnology, MCP enables LLM-based assistants to tap into domain-specific databases and applications – vastly expanding their effective knowledge beyond what they were trained on. This report surveys the **top 20 MCP servers relevant to biotech use cases**, grouped by domain. These servers connect **AI agents** to critical biological and biomedical resources, including scientific literature, genomic/protein databases, chemical/drug databases, clinical trial registries, and more.

Key findings include:

- **Rich interconnected ecosystem:** MCP servers now exist for virtually every major biotech data source. For example, an MCP server for PubMed allows AI agents to search and retrieve biomedical literature programmatically (^[1] www.mcpnow.io), while GOViral (medRxiv, bioRxiv) MCP servers grant access to health-science preprints (^[2] lobehub.com) (mcp.so). On the molecular side, Ensembl and NCBI MCP servers provide genome and gene data (^[3] lobehub.com) (beta.mcp.so), UniProt and AlphaFold servers deliver protein sequences and structures (^[4] lobehub.com) (^[5] github.com), and Reactome/STRING servers supply pathway and protein-interaction networks (^[6] mcpservers.org) (^[7] mcpservers.org). Chemical and drug discovery are covered by ChEMBL, SureChEMBL, PubChem, and OpenFDA servers (^[8] github.com) (^[9] github.com), while clinical trials data is accessible via a ClinicalTrials.gov MCP server (^[10] github.com). Aggregator servers (e.g. Biotoools, BioThings) wrap multiple resources into one API (^[11] lobehub.com) (^[12] lobehub.com).
- **Capabilities:** Each MCP server exposes a set of specialized “tools” (functions) for that domain. For instance, the PubChem MCP server by Cyanheads provides a suite of ten tools for chemical structure searches, property retrieval, similarity searches, and bioassay queries (^[13] github.com). Similarly, the ChEMBL MCP Server offers 22 tools for compound search, target analysis, activity assays, and drug-status queries (^[8] github.com). The Ensembl MCP Server exposes gene/transcript lookup, sequence retrieval, homology finding, variant annotation, regulatory-region queries, and more (25 tools total) (beta.mcp.so). A “Biotoools” MCP server integrates literature (PubMed), UniProt, GenBank, KEGG, PDB and analysis tasks into one unified API (^[11] lobehub.com) (^[14] lobehub.com). These rich toolsets mean that an AI agent can – for example – search PubMed **and** fetch protein annotations or structure files all in one conversation.
- **Use cases and impact:** By linking LLMs with curated databases, MCP servers transform how biotech R&D is done. LLMs can now automatically gather and synthesize information from primary data sources. For example, an AI assistant could query the PubMed MCP server to **automate literature reviews** or hypothesis generation (^[1] www.mcpnow.io), retrieve up-to-date clinical trial data via the ClinicalTrials.gov MCP server (^[10] github.com), and fetch drug properties from OpenFDA and ChEMBL servers to streamline **drug discovery**. Early studies confirm that LLMs excel at mining biomedical literature (^[15] www.mdpi.com), and with MCP integration their outputs become **evidence-based and actionable**. In practice, enterprises are already building MCP frameworks; e.g. Workato’s new platform lets AIs operate securely on business data including FDA labeling and medical records (^[16] www.axios.com).
- **Security and governance:** The open MCP model also raises critical challenges. Researchers have identified threats such as **identity fragmentation**, where inconsistent authentication across MCP services can expose sensitive data (^[17] www.techradar.com). A real-world incident saw a malicious MCP server (masquerading as the “Postmark” email service) exfiltrate confidential emails from organizations that installed it (^[18] www.itpro.com). These incidents underscore the need for robust validation, monitoring, and least-privilege design in MCP deployments. As one report notes, although >15,000 MCP servers exist globally (^[17] www.techradar.com), the community must adopt strong governance (e.g. token-based access controls, certified server registries) to mitigate such risks.

- **Future outlook:** The MCP landscape is rapidly evolving. New standards, tools, and certifications are emerging to ensure interoperability and trust. Enterprise solutions promise turnkey MCP management (^[16] www.axios.com). In biotechnology, we expect MCP to accelerate drug discovery pipelines (by connecting AI with cheminformatics, bioassays, etc), genomic medicine (linking AI with variant databases and clinical registries), and fundamental research (AI-assisted analysis of new datasets). However, this power comes with responsibilities: organizations must carefully vet MCP servers and continuously audit agent actions. With proper safeguards, MCP can unlock a new era where **AI agents and biotech data work together seamlessly**, dramatically speeding research, improving reproducibility, and enabling discoveries unsupported by either humans or AI alone.

The following sections detail background, server by server analysis, evidence, case examples, and implications for the future of MCP in biotech.

1. Introduction and Background

Advances in artificial intelligence have revolutionized many fields, and **biotechnology** is no exception. Machine learning has powered breakthroughs in genomics, [drug discovery](#), and clinical diagnostics. Recently, **generative AI** (large language models, LLMs) has shown exceptional ability to understand and generate human-like text. However, LLMs typically have fixed “knowledge” based on their training data, and lack reliable access to updated or specialized domain data. This limitation has spurred development of [AI agent frameworks](#) (e.g. LangChain) that allow chaining LLM reasoning with external tools and databases. Yet until recently, there was no standard way for LLMs to discover and safely invoke external knowledge sources; integration typically required bespoke engineering.

In 2025, a new open standard called the **Model Context Protocol (MCP)** emerged to address this gap (^[19] www.axios.com) (^[20] www.techradar.com). MCP (sometimes called “Machine Communication Protocol” in fintech contexts (^[21] www.techradar.com)) is a *vendor-neutral*, transport-independent protocol designed for AI models to access tools and data. As Axios reports, Anthropic developed MCP to provide “a relatively simple and standardized way for developers to integrate AI into everyday tools” (^[19] www.axios.com). In essence, MCP defines how an AI assistant (the *client*) discovers available services and invokes them via well-defined “tool calls,” with JSON arguments and structured outputs. This allows LLMs to call APIs by name (e.g. `search_pubmed(query)`) rather than via raw text or ad-hoc code.

Several technology firms and industry leaders have embraced MCP and similar ideas. TechRadar Pro notes that MCP is “explorer a crucial standard for integrating AI systems with tools, APIs, and data sources” (^[22] www.techradar.com). Stripe and Adyen (fintech) even announced a “Machine Communication Protocol” in banking, showing MCP’s appeal beyond tech companies (^[21] www.techradar.com). The goal is a **composable AI architecture**: LLMs become modular agents that can autonomously invoke domain-specific tools. This removes the need for manual interface coding: e.g., an editor could simply instruct an AI, “Find all clinical trials on gene X,” and the agent would call the [ClinicalTrials.gov](https://clinicaltrials.gov) MCP server to fetch results (^[10] github.com).

MCP’s rise is significant. Estimates suggest **>15,000 MCP servers** have been deployed globally by late 2025 (^[17] www.techradar.com). Major cloud and AI vendors are building ecosystems around MCP. For instance, Workato launched an enterprise MCP platform that can connect ChatGPT, Claude, Amazon Q, Google Gemini, etc., to enterprise data sources in a governed way (^[16] www.axios.com). At the same time, critiques have emerged: TechRadar cautions about security loopholes (identity fragmentation) in naive MCP implementations (^[17] www.techradar.com), and cases of malicious MCP modules have already been reported (^[18] www.itpro.com). These issues highlight that while MCP opens powerful capabilities, it also requires careful management.

In biotechnology specifically, the ability to harness MCP is poised to be transformative. Biotech R&D relies on vast specialized databases (genes, proteins, structures, pathways, chemical libraries, clinical data), and MCP provides a uniform way to plug these into AI workflows. Rather than having scientists manually query each database, or write custom code to interface with them, an AI assistant can automatically invoke the appropriate

MCP server tool. For example, an AI-driven hypothesis generator might pull background information on a protein from UniProt and recent literature from PubMed in one combined query, then design experiments accordingly. Recent reviews confirm that LLMs integrated into literature discovery pipelines greatly improve hypothesis generation and data extraction in biomedicine (^[15] www.mdpi.com). MCP completes the picture by connecting these LLMs to the underlying data sources.

This report provides a **comprehensive analysis of the top 20 MCP servers** pertinent to biotech. We first survey different categories of biological data and the corresponding MCP servers, then dive into specific servers, their capabilities, statistics, and usage examples. We include both open-source community servers (often on GitHub or listed in MCP directories) and emerging enterprise offerings. Each server is evaluated for its relevance to biotech tasks. We also discuss case studies and scenarios illustrating how these servers can be used together. Finally, we analyze security considerations, governance implications, and future directions for MCP in the biotech domain. Throughout, every claim is backed by credible sources to ensure a factual and unbiased presentation.

2. MCP in Biotech: Data and Use-Case Categories

Biotech research spans numerous data domains. Table 1 below categorizes these domains and lists representative MCP servers for each. We group the **Top 20 Bio-MCP servers** into logical categories:

Category	MCP Servers (Examples)	Key Data/Function
Scientific Literature & Preprints	PubMed (cyanheads), medRxiv, bioRxiv (JackKuo)	Indexed biomedical articles and preprints; search and retrieve papers, abstracts, metadata. (^[1] www.mcpnow.io) (^[2] lobehub.com)
Genome & Gene Databases	Ensembl (Augmented_Nature), NCBI Datasets (Aug_Nature), GTEx (Aug_Nature)	Genome sequences, gene/transcript info, gene expression by tissue, eQTLs, variant annotations. (^[3] lobehub.com) (^[23] lobehub.com)
Gene Ontology (Functional Data)	Gene Ontology (Augmented_Nature)	Controlled vocabulary for gene function. Search terms, navigate ontology tree, retrieve gene annotations. (^[24] glama.ai)
Protein Databases & Structures	UniProt (Aug_Nature), Protein Data Bank (RCSB PDB server), AlphaFold (Aug_Nature)	Protein sequences/annotations (UniProt), 3D structures (PDB & AlphaFold predictions), confidence analysis. (^[4] lobehub.com) (^[5] github.com)
Pathways & Networks	Reactome (Augmented_Nature), STRING (MCPmed)	Biological pathways and protein interaction networks (Reactome); PPI networks and enrichment (STRING). (^[6] mcpservers.org) (^[7] mcpservers.org)
Chemicals & Drug Data	PubChem (cyanheads), ChEMBL (Aug_Nature), SureChEMBL (Aug_Nature), OpenFDA (Aug_Nature)	Chemical compounds' properties and bioactivities (PubChem, ChEMBL); patent chemistry (SureChEMBL); drug labels/adverse events (OpenFDA). (^[13] github.com) (^[8] github.com)
Clinical Trials & Public Health	ClinicalTrials.gov (cyanheads)	Clinical study metadata (design, status, conditions); search by disease or intervention. (^[10] github.com)
Expression Data	GEO (MCPmed)	Gene expression datasets (microarrays, RNA-Seq) from NCBI GEO; search datasets by keywords. (^[25] lobehub.com)

Category	MCP Servers (Examples)	Key Data/Function
Integrated & Utility Servers	Biotoools (BACH-AI-Tools), BioThings.io (Aug_Nature)	Meta-servers wrapping multiple resources: Biotoools covers PubMed/UniProt/GenBank/KEGG/PDB ([11] lobehub.com); BioThings (MyGene/MyVariant) for gene/variant annotation ([12] lobehub.com).

Table 1. Major biotech data categories and corresponding MCP servers. Citations for examples are provided in text.

Each category plays a crucial role:

- Literature and Preprints:** The **PubMed MCP Server** provides AI access to ~36 million PubMed citations. Its tools (search, fetch abstracts, related-articles) enable automated literature review ([1] www.mcpnow.io). Similarly, specialized MCP servers for **medRxiv** and **bioRxiv** (health and biology preprint archives) let AI agents query the latest unpublished research ([2] lobehub.com) (mcp.so). These servers allow rapid scouting of current findings, beyond what's in the static training corpus.
- Genomic Data:** Databases like **Ensembl** and **NCBI** contain extensive genomic and gene information. For example, the **Ensembl MCP Server** offers tools to lookup gene/transcript details, retrieve sequences, find orthologs, analyze variants, etc. (beta.mcp.so). **NCBI Datasets MCP** provides genome assemblies, gene models, taxonomy, and more (31 tools covering sequences, genes, taxon) ([3] lobehub.com). The **GTEx MCP Server** gives access to human tissue-specific gene expression and eQTL data (25 tools for expression analysis) ([23] lobehub.com). These servers enable AI agents to retrieve up-to-date genomic annotations and expression patterns, crucial for precision medicine and genetic research.
- Gene Ontology (GO):** The **GO MCP Server** exposes the entire Gene Ontology – a structured vocabulary of biological functions – to AI. With tools to search terms, validate IDs, and fetch GO annotations, GPT assistants can perform functional enrichment or term lookups as part of analysis ([24] glama.ai). This bridges language-based AI reasoning with structured biology semantics.
- Proteins and Structures:** **UniProt** is the canonical repository of protein sequences and annotations. Its MCP server (7 tools) allows searching proteins by name or gene and retrieves full annotation entries ([4] lobehub.com). Structural data is served by the **RCSB PDB MCP Server**, which lets agents query PDB entries, download structure files in PDB or mmCIF format, and fetch polymer component info ([26] lobehub.com) ([27] lobehub.com). The cutting-edge **AlphaFold MCP Server** connects to the AlphaFold structure database, supporting batch structure retrieval, sequence searches, and confidence scoring ([5] github.com). Together, these servers enable LLMs to incorporate detailed protein-level knowledge and 3D data in real-time analyses.
- Pathways and Networks:** The **Reactome MCP Server** provides access to curated pathways – AI tools can search pathways by keyword, retrieve pathway details, map genes to pathways, and explore interactions ([6] mcpservers.org). The **STRING-MCP** server (STRING database) lets agents obtain protein–protein interaction networks and perform network analysis at scale ([7] mcpservers.org). These allow AI assistants to reason about functional modules and molecular interactions underlying phenotypes.
- Small Molecules and Drugs:** **PubChem's** MCP server (10+ tools) makes chemical entities and bioassays queryable by AI ([13] github.com). **ChEMBL** (a large bioactivity database) has an MCP API with 22 tools for drug discovery workflows (compound/target search, activity data, clinical status) ([8] github.com). **SureChEMBL** covers patent(IP) data with tools for searching patent literature by compound or text ([28] github.com). Finally, the **OpenFDA MCP Server** taps FDA's public drug and device datasets (labels, adverse events, recalls) with tools for comprehensive safety and recall queries ([9] github.com). Together, these empower AI agents to support cheminformatics, pharmacovigilance, and regulatory research.
- Clinical Trials:** The **ClinicalTrials.gov MCP Server** offers programmatic access to the NIH clinical-trials registry. Its tools enable searching studies by condition, intervention, sponsor, geography, etc., and fetching full trial records ([10] github.com). This allows LLMs to automatically incorporate clinical evidence about diseases and interventions into their reasoning.

- **Gene Expression and Other:** Expression data from NCBI GEO can be accessed via a GEO MCP server (^[25] [lobehub.com](#)), allowing search of expression datasets by keyword or gene. (Similar servers could connect to ArrayExpress or SRA.) In addition, utility servers like **Biotools** and **BioThings** aggregate multiple resources to reduce fragmentation (^[11] [lobehub.com](#)) (^[12] [lobehub.com](#)).

Importantly, these servers are typically provided as installable modules (some on GitHub/npm) that run locally or on a network. An AI agent configured for MCP will have a list of server commands it can invoke. As a result, one can build pipelines where an LLM simply issues natural-language prompts that cause it to call these tools. For example, instead of asking "What is the UniProt entry for P04637?", a GPT agent could format a tool call like `get_protein_info(accession="P04637")` to the UniProt MCP server (^[4] [lobehub.com](#)), and then incorporate the returned JSON into its answer. CPython or Claude can make these calls through STDIO or HTTP as needed.

This tight coupling of LLM and data sources marks a significant advance. Recent biomedical literature reviews highlight that **LLMs inherently excel at processing vast text corpora and uncovering hidden links** (^[15] [www.mdpi.com](#)). MCP servers extend this capability by giving LLMs the ability to **fetch the actual data** – enabling evidence-backed inferences. For example, Taleb *et al.* (2024) show that integrating GPT-based LLMs into literature-based discovery (LBD) workflows greatly enhances hypothesis generation and scalability compared to traditional methods (^[15] [www.mdpi.com](#)). In practice, a biotech researcher using an MCP-enabled AI could ask, "Find recent trials of drug Y in acute myeloid leukemia and summarize their outcomes." The agent could call the ClinicalTrials server, retrieve relevant trial records, then query PubMed or NIH Bookshelf via LTC queries (or the Healthcare MCP server (^[29] [github.com](#))) for related publications, all automatically.

In the sections that follow, we examine the **capabilities and details of each top MCP server**, organized by the domains above. We provide data-driven analysis (e.g. tool counts, dataset sizes) and cite documentation for each. Wherever possible, we include usage examples and potential case studies illustrating how these servers can transform workflows in biotech, pharma, and research.

3. MCP Servers for Literature and Preprints

A. PubMed MCP Server. PubMed is the flagship literature database for biomedical research, housing over 36 million citations (MEDLINE and related journals). The *PubMed MCP Server* (by developer **cyanheads**) provides AI tools to fully utilize PubMed. Per its documentation, this server "empowers AI agents... with comprehensive access to PubMed" via NCBI's E-utilities (^[1] [www.mcpnow.io](#)). Core tools include:

- `pubmed_search_articles(query)` : Search PubMed with filters, returning metadata for matching articles.
- `pubmed_fetch_contents(pmid)` : Retrieve full details (title, abstract, authors, MeSH terms) for a specific PMID.
- `pubmed_article_connections(pmid)` : Find related citations (cited-by, similar articles, MeSH) or format a citation.
- `pubmed_generate_chart(data)` : Create visual charts from query results.
- `pubmed_research_agent(components)` : Assist in drafting a structured research plan outline.

These tools allow an LLM to automate tasks like literature review. For instance, to gather background on "CRISPR gene editing in cancer", an AI could invoke `pubmed_search_articles("CRISPR AND cancer AND 2023", filters=...)` to get recent studies, then use `get_publication_abstract` to read abstracts of the most relevant ones. All returned data is structured JSON for easy consumption by the model.

According to the server README (^[1] [www.mcpnow.io](#)), this MCP is "production-grade" with modular design and robust error handling. Example usage suggests it can even generate plots (`pubmed_generate_chart`) of

publication counts or topics over time. Such AI-augmented literature mining could accelerate systematic reviews or monitor emerging trends.

B. medRxiv MCP Server. medRxiv hosts health-sciences preprints. The *medRxiv MCP Server* (JackKuo666) enables search of medRxiv articles via an MCP interface (^[2] lobehub.com). Its core features include "paper search with keywords", rapid metadata retrieval by DOI, and even local caching of PDFs. With 20,000+ health preprints (COVID-19, epidemiology, etc.), an AI agent can use this to access the very latest unpublished research. The server's features ("🚀 Efficient Retrieval", "Paper Search", "Paper Access") indicate it returns both metadata and PDF text (^[30] lobehub.com). This can feed hypotheses (e.g., an AI can read upcoming vaccine trial results directly from medRxiv).

C. bioRxiv MCP Server. bioRxiv provides biology preprints. The *bioRxiv MCP Server* (JackKuo666) similarly allows LLMs to query bioRxiv content through tool calls (mcp.so). Key features are paper keyword search, fast metadata lookup, and local paper caching. As one MCP listing notes, it "enables AI assistants to search and access bioRxiv papers through a simple MCP interface" (mcp.so). Use cases include retrieving cutting-edge findings that haven't made it to journals yet – for example, in genomics or molecular biology. By plugging both medRxiv and bioRxiv into MCP agents, an AI can consider an expanded literature arena that includes pre-publication results.

Discussion (Literature MCPs): Scientific knowledge is rapidly evolving, especially in fast-moving fields like genomics and immunology. The combination of PubMed, medRxiv, and bioRxiv MCP servers ensures that an AI assistant has up-to-date literature access. For example, one can imagine an automated "alert" agent that periodically searches PubMed plus medRxiv/bioRxiv for new papers matching specific biomedical keywords, then compiles summaries. Recent research suggests LLMs excel at identifying novel insights when integrated into literature discovery tasks (^[15] www.mdpi.com). MCP servers will let these AI's findings be grounded in actual data rather than hallucinations.

Potential pitfalls include the risk of bias or misinformation if a preprint has errors. Thus robust verification (cross-checking multiple sources) is advisable. Security is less of a concern here since these are public resources. However, rate-limiting and API usage policy (e.g. NCBI's E-utility limits) must be respected by MCP clients; most server implementations incorporate key or tooling to handle this (^[1] www.mcpnow.io).

4. Genomic and Gene MCP Servers

A. Ensembl MCP Server. Ensembl is a premier genome browser and annotation database (human and many other species). The *Ensembl MCP Server* developed by **Augmented Nature** provides a rich interface to Ensembl's REST API (beta.mcp.so). It offers 25 *specialized tools* covering:

- **Gene & Transcript Info:** Tools like `lookup_gene(id_or_symbol)` and `get_transcripts(gene_id)` retrieve detailed gene models, transcripts, exons, UTRs, etc. (beta.mcp.so).
- **Sequence Data:** `get_sequence(region_or_feature)` can extract genomic DNA or mRNA sequence; `get_cds_sequence(transcript_id)` retrieves coding sequences; `translate_sequence` gives protein translations (beta.mcp.so).
- **Comparative Genomics:** `get_homologs(gene)` finds orthologs/paralogs across species; `get_gene_tree(gene_family_id)` retrieves phylogenies (beta.mcp.so).
- **Variant Data:** `get_variants(region, assembly)` returns known variants; `get_variant_consequences(variant_id)` predicts variant effects on transcripts (beta.mcp.so).
- **Regulatory Annotations:** Tools to fetch enhancers, promoters, motifs in a region (e.g. `get_regulatory_features`) and transcription-factor binding motifs (beta.mcp.so).

- **Cross-References & Assemblies:** `get_xrefs(gene)` maps to external DBs; `map_coordinates()` converts between assemblies (beta.mcp.so).
- **Batch Processing:** Tools for bulk gene lookups or sequence fetches (beta.mcp.so).

In total, the Ensembl MCP server covers almost **all aspects of Ensembl data**. For example, an AI assistant tasked with analyzing a genomic locus (say "APOE region in human chromosome 19") could call `get_variants("19:44900000-44905000")` to list SNPs in that region, then `get_gene("APOE")` to fetch gene/exon structure. It could even translate sequences or search for homologs (e.g. "Find mouse ortholog of human APOE"). The server supports multiple species and defaulting to human if not specified. Advanced example usage shows commands like `translate this DNA sequence to protein` being automatically mapped to the proper MCP calls (beta.mcp.so).

B. NCBI Datasets MCP Server. The NCBI offers the *Datasets* API, which provides programmatic access to genome assemblies, sequences, taxonomy, and more. The *Unofficial NCBI Datasets MCP Server* (Augmented Nature) exposes **31 tools** covering genomics and taxonomy (^[31] lobehub.com). The features include:

- **Genome Access:** Tools to search available species, list assemblies, fetch genome FASTA and annotation (e.g. `get_genome_info(accession)`, `get_genome_summary(accession)`).
- **Gene/Protein Data:** Fetch gene details (name, location, function), protein sequences, and their function annotations.
- **Taxonomy:** Query taxonomic lineages, search species by name.
- **Utility:** Provide quick search across NCBI databases for genes or proteins and their cross-links.

For example, an AI agent could call `search_genome("Homo sapiens")` to list human reference assemblies, then `get_gene("BRCA1")` to retrieve all known metadata about the BRCA1 gene from Entrez. The server handles NCBI's rate limits (using an optional API key). It essentially brings the power of NCBI's backend to MCP. This is valuable for integrative queries that span multiple NCBI resources.

C. GTEx MCP Server. The Genotype-Tissue Expression (GTEx) project has massive multi-tissue expression data. *Unofficial GTEx Portal MCP Server* (Augmented Nature) is designed to query GTEx via MCP. It offers **25 tools** for tasks such as:

- Searching expression profiles of genes across tissues.
- Finding expression quantitative trait loci (eQTLs) for specific gene-variant pairs.
- Retrieving tissue-level gene/variant statistics.
- Bulk queries for sets of genes/variants.

This means an AI assistant can ask, e.g., "Which tissue has highest expression of gene X?" and the assistant can call `get_tissue_expression("GSTM5")` to obtain TPM values, or `get_eQTLs(gene="TMZ", snp="rs...")` to identify regulatory variants. Such queries are critical in functional genomics and personalized medicine. By integrating GTEx via MCP, research on genotype-phenotype associations becomes directly accessible to AI analysis.

Discussion (Genomic Servers): These genomic MCP servers enable AI agents to obtain **raw biological data** that was previously behind complex databases. For example, Curated pipelines often rely on Ensembl or NCBI data, but pre-MCP required writing API calls or SQL. MCP abstracts that; a researcher can now instruct an AI in plain language. Moreover, because Ensembl and NCBI are regularly updated, the AI always sees the latest reference builds and annotations.

For biotech use, this has many applications: e.g., variant interpretation in genomic medicine. An AI can combine Ensembl's variant impact calls with GTEx expression patterns and GO functional data to assess a mutation's

significance. Another example: in synthetic biology, an AI agent could design coding sequences by calling `get_cds_sequence` on Ensembl and then analyzing codon usage. By directly querying the authoritative sources, answers stay accurate and reproducible.

Challenges include ensuring correct chromosome nomenclature and reference versions. MCP clients must specify assemblies (e.g. GRCh38 vs GRCh37). Some servers (like Ensembl) default to a particular build (often GRCh38) and allow mapping (beta.mcp.so), but ambiguous input could lead to cross-mapping errors. Governance is less worrying here since these are generally public data, but data privacy might be a factor if querying personal genome repositories (not covered here).

Overall, genomic MCP servers greatly enhance AI's ability to handle DNA/protein level questions. They supply the factual backbone for biomedical queries, complementing the literature MDPI review's insight that LLMs can generate hypotheses by combing massive data (^[15] www.mdpi.com). Now, the AI can *fetch* that data directly.

5. Gene Ontology (GO) MCP Server

The *Gene Ontology (GO) MCP Server* (Augmented Nature) provides AI access to the structured vocabulary of biological terms. GO covers three ontologies: molecular function, cellular component, and biological process. This MCP server offers tools such as:

- `search_go_terms(query)` : find GO term IDs by keyword across ontologies.
- `get_go_term(id)` : retrieve full information for a GO term (name, namespace, definition, parents/children).
- `get_gene_annotations(gene_id)` : fetch all GO annotations for a given gene or protein (via UniProt or other gene IDs).
- `validate_go_term(id)` : check if a GO ID is valid.

Using this server, an AI assistant can perform ontology-driven analysis. For example, to analyze a gene list, the assistant could call `get_go_term("GO:0006915")` to get details for apoptosis, or annotate a protein by invoking `get_go_term` on all of its GO IDs found via UniProt. One tool might even return a list of related terms (parents/children) to explore associated processes.

In practical terms, this lets LLMs incorporate functional knowledge explicitly. A biotech Q&A could be: "*Which molecular functions are overrepresented in my gene set?*" – the agent could gather GO terms via `get_gene_annotations`, then perform statistical enrichment with another tool. This adds semantic depth to the AI's reasoning.

GO annotations are continually updated by consortia like the GO Consortium, so the MCP server ensures AI answers reflect current biology. For example, if a protein's function is newly characterized, that appears in the GO annotation fetch. This dynamic connection contrasts with static LLM knowledge which might be out-of-date.

6. Protein and Structure MCP Servers

A. UniProt MCP Server. *Unofficial UniProt MCP Server* (Augmented Nature) gives AI agents advanced access to the UniProtKB protein sequence database (^[4] lobehub.com). Key endpoints include:

- `search_proteins(query, organism)` : Queries UniProt by protein name, keyword, or complex search, returning accession IDs and brief info** (^[4] lobehub.com).
- `get_protein_info(accession)` : Fetches full UniProt entry for one accession: sequence, function, subcellular location, domains, variants, cross-references to 150+ databases (^[32] lobehub.com).

- `search_by_gene(gene_symbol, organism)` : Find proteins by their gene symbols.
- `get_protein_sequence(accession)` : Retrieve the amino acid sequence (FASTA) for an accession.

For example, an AI asked *"What is the function of human CFTR protein?"* could call `search_by_gene(gene="CFTR", organism="human")` to get the accession (P13569), then `get_protein_info("P13569")` to retrieve the complete annotation, including disease relevance, variants, etc. The UniProt MCP thus serves as the definitive source of protein knowledge.

B. RCSB PDB MCP Server. Structural biology data lives in the Protein Data Bank. The *RCSB PDB MCP Server* (by **cnyambura**) provides tools to interact with the RCSB API (^[33] [lobehub.com](https://github.com/cnyambura/lobehub.com)). Available calls include:

- `get_pdb_entry(pdb_id)` : Get metadata about a structure (title, method, resolution, authors, source organism) (^[34] [lobehub.com](https://github.com/cnyambura/lobehub.com)).
- `get_polymer_entity(pdb_id, entity_id)` : Retrieve sequence and type information for a specific polymer in the structure (protein/DNA chain) (^[35] [lobehub.com](https://github.com/cnyambura/lobehub.com)).
- `download_structure_file(pdb_id, format, output_dir)` : Download the 3D coordinate file in PDB, mmCIF, or XML format (^[36] [lobehub.com](https://github.com/cnyambura/lobehub.com)).
- `query_rcsb_api(endpoint, params)` : A generic call to any RCSB API path (e.g. get assembly, ligand info) (^[37] [lobehub.com](https://github.com/cnyambura/lobehub.com)).
- `search_pdb_by_organism(organism)` : Helper to build queries for specific organisms (^[38] [lobehub.com](https://github.com/cnyambura/lobehub.com)).

With these, an AI can fetch actual macromolecular structures. Example use: *"Fetch the atomic structure for HBB (hemoglobin beta) and provide chain sequences."* The assistant calls `get_pdb_entry("4HHB")` to verify that's human hemoglobin, then `get_polymer_entity("4HHB", "1")` and `get_polymer_entity("4HHB", "2")` for the two chains. It could then instruct `download_structure_file("4HHB", "cif", "/tmp")` to retrieve coordinates for analysis or visualization. This opens the door to AI-assisted structural biology: predicting ligand interactions, explaining structural motifs, or even generating images (the `download_structure_file` returns a file path that the assistant could post-process via plotting tools).

C. STRING MCP Server. Protein interaction networks can elucidate function. The **STRING-MCP** (developed by MCPmed) provides Python-based MCP to query the **STRING** database (^[7] [mcpservers.org](https://github.com/mcpmed/mcpservers.org)). Key methods include `map_identifiers()` to map gene names to STRING IDs and `get_network_interactions()` to retrieve the PPI network for a set of proteins (^[39] [mcpservers.org](https://github.com/mcpmed/mcpservers.org)). An AI agent can thus ask *"Show me the interaction network for TP53 and its nearest neighbors."* It would call `get_network_interactions(gene="TP53_human", threshold=0.7)` and receive a graph or JSON of interacting proteins. This complements the PDB data by situating proteins in their cellular network context.

D. AlphaFold MCP Server. The AlphaFold Protein Structure Database holds predicted structures for nearly every protein in UniProt. The *AlphaFold MCP Server* (Augmented Nature) allows retrieval and analysis of these predictions (^[5] [github.com](https://github.com/AugmentedNature/alpha-fold-mcp-server)). Its features include:

- **Structure Retrieval:** Fetch predicted 3D models by UniProt ID (human or other species) in PDB/CIF/JSON formats (^[40] [github.com](https://github.com/AugmentedNature/alpha-fold-mcp-server)).
- **Search & Discovery:** Search for proteins by name or get lists of structures for an organism (^[41] [github.com](https://github.com/AugmentedNature/alpha-fold-mcp-server)).
- **Confidence Analysis:** Tools to obtain per-residue confidence scores (pLDDT) and highlight low-confidence regions (^[42] [github.com](https://github.com/AugmentedNature/alpha-fold-mcp-server)).
- **Batch Mode:** Download many structures or analyze them in bulk (^[43] [github.com](https://github.com/AugmentedNature/alpha-fold-mcp-server)).
- **Structure Similarity:** Compare structures or find similar folds (^[44] [github.com](https://github.com/AugmentedNature/alpha-fold-mcp-server)).

For instance, an AI assistant can request `get_structure("P01308", format="pdb")` to retrieve the AlphaFold model of insulin, and call a tool to highlight low-confidence loops. This server effectively brings *de novo* structural predictions into the AI knowledge base, complementing experimental PDB data.

Discussion (Protein/Structure MCPs): These servers empower biotechnology AI with protein-level insight. Tasks like mutation effect prediction, enzyme mechanism explanation, or antigen design benefit from them. For example, in antibody engineering, the AI can fetch antigen structures (AlphaFold/PDB) and propose binding interface mutations. In diagnostics, LLMs can combine PDB structural data with pathway context (STRING, Reactome) to pinpoint mutation hotspots.

From a reliability standpoint, UniProt and PDB are highly curated; AI can trust those annotations. AlphaFold structures, while high-quality, have variable confidence – which is why their MCP server includes confidence tools. Agents should communicate uncertainty when relying on predicted regions. Security is not a major concern here (these are public scientific data), but correct parsing of file outputs (e.g. ensuring coordinate files are not incorrectly interpreted) is important. Notably, GPT-4 and similar models have been shown to assimilate protein structure data when in textual form, so MCP-mediated access could enhance that capability in practice.

7. Pathway and Network MCP Servers

A. Reactome MCP Server. *Reactome* is an expert-curated knowledgebase of biological pathways (metabolic, signaling, etc.). The **Unofficial Reactome MCP Server** (Augmented Nature) provides 8 verified tools for interacting with Reactome's live API (^[6] mcp-servers.org). These include:

- **Pathway Search** 🔍 : Find pathways by keyword; returns pathway IDs and descriptions.
- **Pathway Details** 📊 : Get comprehensive information on a pathway, including its constituent reactions and participating molecules.
- **Gene-to-Pathways** 🧬 : List all pathways that involve a given gene/protein.
- **Disease Pathways** 🦠 : Search pathways associated with a disease or genetic condition.
- **Hierarchy** 🌲 : Navigate parent/child relationships among pathways.
- **Pathway Participants** 📋 : List all proteins/compounds participating in a pathway.
- **Reaction Info** 📄 : Get details for biochemical reactions within a pathway.
- **Protein Interactions** 🔗 : Although Reactome is not a PPI database, this tool returns molecules that interact within reaction context.

For example, an AI could query `pathway_search("TCA cycle")` to identify the central carbon metabolism pathway, then call `pathway_details(id)` to retrieve all enzymes and compounds involved. Alternatively, `gene_to_pathways("BRCA1")` would list DNA-repair and cell-cycle pathways involving BRCA1. This bridges gene-centric analysis to systems-level understanding. An AI tasked with drug repurposing might use Reactome tools to find pathways overlapping a drug target and a disease gene, highlighting novel intervention points.

Discussion (Pathway Servers): Pathways contextualize molecular data into cellular processes. By combining Reactome with MCPs like Ensembl or UniProt, an agent can map genotype to phenotype. For instance, given a list of patient mutations in metabolic genes, an AI could identify which metabolic pathways are disrupted and suggest dietary or drug interventions. However, pathway data is complex; errors can arise if a term maps to multiple IDs or if an entity has synonyms. Careful use of gene/protein identifiers (e.g. HGNC symbols or Ensembl IDs) is needed. Reactome MCP returns reliable, curated data (Reactome is a well-established BioCyc-like resource).

In the broader pipeline, Reactome fosters **explainability**: an LLM can show not just "X gene is downregulated" but "which pathway X normally participates in and what downstream effects that may have." Surveys of LLM in bioinformatics emphasize that such context (pathways, networks) greatly aids interpretation (^[15] www.mdpi.com).

8. Chemical and Drug MCP Servers

A. PubChem MCP Server. The **PubChem MCP Server** (cyanheads) taps into PubChem's PUG REST API to deliver an extensive suite of chemical tools (^[13] github.com). PubChem contains 110+ million compounds with structures and bioassay results. This MCP server's tools include:

- `pubchem_search_compound_by_identifier(type, value)`: Find compound IDs by names, SMILES, InChIKey (^[45] github.com).
- `pubchem_fetch_compound_properties(cids, properties)`: Get physical/chemical properties (molecular weight, LogP, formula) for given CIDs.
- `pubchem_get_compound_image(cid, size)`: Render 2D structure image.
- `pubchem_search_compounds_by_structure(type, query, max)`: Substructure or similarity search.
- `pubchem_search_compounds_by_formula(formula)`: Find compounds matching an element formula.
- `pubchem_fetch_substance_details(sid)`: Retrieve details by Substance ID (SID).
- `pubchem_search_assays_by_target(target_type, query)`: Find BioAssays related to a target (protein name, gene).
- `pubchem_fetch_assay_summary(aid)`: Get assay information by assay ID.
- `pubchem_fetch_compound_xrefs(cid)`: List external database links for a compound.

These enable chemical informatics tasks. For instance, an AI could do

```
pubchem_search_compound_by_identifier("name", "Aspirin") to get CID 2244, then
pubchem_fetch_compound_properties([2244], ["MolecularWeight", "XLogP"]) . It could search related analogs
via pubchem_search_by_similarity . A drug-discovery use case: given a protein target from a prior MCP call
(e.g. UniProt search), the agent might use pubchem_search_assays_by_target("ProteinName", "EGFR") to gather
known assay data for EGFR inhibitors, then analyze activities with `
```

Because PubChem is public, there are no special access restrictions, but the volume of data demands efficient API usage. The MCP server handles batching and JSON output. Data quality is high – PubChem's integrated curation from NIH-supported sources ensures reliable compound data.

B. ChEMBL MCP Server. **ChEMBL** (by EBI) is a curated bioactivity database (~2 million compounds with drug targets/assays). The *ChEMBL MCP Server* (Augmented Nature) provides 22 tools grouped into chemical search, target analysis, assay data, and drug status (^[8] github.com). Highlights include:

- **Chemical search:** Look up compounds by name or ID, fetch structure formats (SMILES, SDF), similarity searches.
- **Target analysis:** Search targets (proteins) by name, fetch target info, retrieve compounds tested on a target, and even find pathways involving the target (^[46] github.com).
- **Bioactivity search:** Query assay results (`activity_search`) by potency range, retrieve detailed assay protocols (`Detailed Assay Info`), dose-response data.
- **Drug development:** Identify which compounds are approved drugs versus research compounds, fetch drug development status and clinical trial links (^[47] github.com).

For example, an AI agent looking at a molecule identified in a gene screen could use ChEMBL:

`chembl_search_compound("PubChem:cid:2244")` to retrieve the compound entry for aspirin, then `chembl_activity_search(cids= [2244])` to see all target activities. Or, given a protein target name, `target_compounds("EGFR")` would list all known inhibitors and their potencies, guiding repurposing hypotheses. In summary, ChEMBL MCP directly bridges molecules and biology with mechanism-of-action data.

C. SureChEMBL MCP Server. SureChEMBL contains patent chemistry data. The *SureChEMBL MCP Server* (Augmented Nature) focuses on mining patents for chemical information (^[28] [github.com](#)). Its tools include:

- `search_patents(query)` : Full-text search across all patent fields.
- `get_document_content(patent_id)` : Retrieve the full text (including chemical markups) of a patent.
- `search_by_patent_number()` : Fetch by specific patent numbers.
- `search_chemicals_by_name(name)` : Find compounds mentioned in patents.
- `get_chemical_by_id(chem_id)` : Get compound structure and details from SureChEMBL.
- `export_chemicals()` : Bulk export of chemical datasets.
- `analyze_patent_chemistry()` : Summarize how many chemicals in patents, etc.
- Visualization: `get_chemical_image` renders SMILES, `get_chemical_properties` lists predicted properties (^[48] [github.com](#)).

This allows LLMs to incorporate patent intelligence. For a biotech startup monitoring competitor IP, an AI assistant could call `search_patents("CRISPR AND Cas9")` to find relevant patents, then `get_document_content(id)` to extract example claims. Combined with ChEMBL, one can map novel structures in patents to known bioactivities. SureChEMBL data is a rich source but requires proper attribution. The MCP server's integration makes it programmatically accessible for AI analysis, turning unstructured patents into structured knowledge queries.

D. OpenFDA MCP Server. The U.S. FDA provides open data on drugs, devices, and adverse events via the *OpenFDA API*. The *OpenFDA MCP Server* (Augmented Nature) wraps these datasets (^[49] [github.com](#)). It implements tools for:

- **Drug tools:** Search adverse events reports (`search_drug_adverse_events`), look up drug labels (`search_drug_labels`), query the NDC directory (`search_drug_ndc`), find recalls (`search_drug_recalls`), and search the Drugs@FDA database for product approvals (`search_drugs_fda`) (^[9] [github.com](#)).
- **Device tools:** Analogous queries for medical device 510(k) clearances, classifications, adverse events (MDR), and device recalls (^[50] [github.com](#)).

Each tool supports filters (drug name, manufacturer, date ranges, etc.) to refine results. For instance, an AI writing a drug safety report could call `search_drug_adverse_events("aspirin")` to retrieve recent post-marketing side-effect reports. It might then analyze trends or generate a summary of most frequent adverse events. OpenFDA's rich content (including warnings, recalls) complements ChEMBL data by covering real-world usage and safety.

This FDA MCP is especially relevant in translational biotech and pharmacovigilance. It has been credited for enabling supervised AI coding agents to create real emails and alerts (in fintech context), and here it can empower domain-specific tasks. The server respects the FDA's rate limits and can use an API key for higher throughput (^[51] [github.com](#)). It returns well-structured JSON for all queries.

Discussion (Chem/Drug Servers): Together, PubChem, ChEMBL, SureChEMBL, and OpenFDA close the loop on chemical-biological data. An AI can perform end-to-end analysis: e.g. given a novel compound in a lab, it can search PubChem or ChEMBL for analogs, retrieve properties/bioactivities, scan patents for related compounds

(SureChEMBL), and check regulatory status (OpenFDA). This streamlines drug discovery pipelines. For example, in lead optimization AI workflows, MCP servers let LLMs propose synthetic modifications and immediately query their effects.

The integration of these servers is unprecedented. Traditional computational chemistry required separate scripts for each database; MCP unifies them. However, data volume is huge. PubChem's database is enormous (>100M molecules) so queries must be targeted. ChEMBL's 2M compounds are smaller but heavily annotated. Performance and caching strategies are important: some implementations buffer results (like medRxiv/server caches PDF) to avoid repetition. AI agents should also interpret chemical structure (SMILES) carefully – typically the MCP returns string data, which the LLM can then analyze or compare via subtools.

From a security standpoint, all these are public databases, so risks are low. However, AI outputs must be checked. Since LLMs can hallucinate, any critical conclusions (e.g. "compound X is safe") should be cross-checked by additional queries. Citations from authoritative sources (e.g. citing a PubChem CID and study) increase trust.

9. Clinical Trials and Public Registry MCP Servers

ClinicalTrials.gov MCP Server. *ClinicalTrials.gov* is the primary registry of clinical studies worldwide. The cyanheads *ClinicalTrials.gov MCP Server* provides a developer-friendly interface to their v2 API (^[52] [github.com](#)). Its core tools include:

- `clinicaltrials_search_studies(query, filters)` : Flexible search by combinations of terms (condition, intervention, sponsor, phase, location, etc.), with pagination and sorting.
- `clinicaltrials_get_study(nctid)` : Fetch complete details of a trial by its NCT identifier (or list of NCTs), returning either the concise summary or the full protocol text for each trial.
- `clinicaltrials_analyze_trends(query, by="status")` : Statistically analyze a set of trials, aggregating by attributes like status, country, sponsor, or phase (^[53] [github.com](#)).

For example, an AI agent might call `clinicaltrials_search_studies("oncology", phase=3, status="recruiting")` to fetch current cancer drug trials in Phase III. The returned NCT IDs could then be fed to `get_study` to retrieve official study descriptions, interventions, and outcome measures. Alternatively, one could use `analyze_trends` to quantify how many Phase III trials are recruiting for each cancer type.

This MCP server automates queries that normally require form-filling on the *ClinicalTrials.gov* website or manual API use. It leverages the official API (no extra credentials needed for basic use). The JSON responses include study design, condition, interventions, and contact info, all structured. In biotech R&D, this server enables an AI to incorporate evidence of ongoing research. For instance, when evaluating a drug target, the AI could list relevant active trials and even extract results summaries if available.

NIH Repositories: Beyond *ClinicalTrials.gov*, AI agents can also access other NIH registries via MCP (though not listed among top 20, some hybrid servers do this). For example, Cicatriiz's "Healthcare MCP Server" bundles PubMed and NCBI Bookshelf (textbooks), ICD-10, and DICOM metadata alongside trials and FDA (^[29] [github.com](#)). This illustrates a broader trend: some MCP deployments integrate multiple public health sources together. Such aggregated servers go beyond the scope of our top-20 list but show how MCP can unify the "evidence chain" from patient data (ICD codes) to approved drugs to trials to literature.

Discussion (Clinical MCPs): By connecting to *ClinicalTrials.gov*, AI can keep abreast of what therapies are in human testing. This is crucial in cancer, rare diseases, neurology, etc., where ongoing trials represent cutting-edge. It also enhances regulatory intelligence: identifying trial populations and endpoints helps in drug design. Furthermore, linking trials to PubMed (via NCTID references in publications) is feasible: an AI could cross-

reference NCTs with PubMed MCP (since PubMed records include NCT in metadata). This multi-step link (AI reads trial info → searches literature of trial) exemplifies an intelligent pipeline.

Potential pitfalls: Not all trials yield published results in a timely way. The `analyze_trends` tool is useful to spot representation gaps (e.g. lack of diversity if few countries). Agents must handle very large result sets (hundreds of trials) carefully to avoid overloading memory. Data is relatively clean, and the registry is public domain, so the main concern is semantic accuracy. For example, an AI should distinguish “withdrawn” vs “ongoing” properly, and not infer efficacy without results. The `analyze_trends` tool mitigates data overload by summarizing.

10. Integrated and Utility MCP Servers

While the prior sections covered specialized databases, a couple of MCP servers act as **integrators or utilities** by bundling multiple resources:

A. Biotools MCP Server. Developed by *BACH-AI-Tools*, the **Biotools MCP Server** provides a “comprehensive MCP server for bioinformatics research” (^[11] [lobehub.com](#)). It integrates many of the above data sources into one unified interface. As described, it grants access to “major biological databases and analysis tools including PubMed, UniProt, NCBI GenBank, KEGG, PDB, and more.” (^[11] [lobehub.com](#)). Its available tools (37 total) span numerous categories. We saw examples earlier of its literature search (`search_pubmed`), protein search (`search_uniprot`), nucleotide sequence fetch, GC content analysis, promoter alignment, etc. (^[14] [lobehub.com](#)) (^[54] [lobehub.com](#)). In essence, Biotools is a Swiss Army knife: LLMs can use it to handle a variety of tasks without switching MCP servers. For example, a prompt like “Analyze the sequence features of gene X” could trigger Biotools’ nucleotide ORF finder, promoter aligner, and UniProt interfacing tools all in one go.

Because Biotools is a composite server, its usefulness comes from breadth and convenience. It likely runs locally as one process (e.g. a Python/Node server) with built-in calls to external databases. The potential downside is that performance can be chokepointed by the slowest component. Also, robustness depends on each integration (if PubMed API changes, one part fails). However, from a user’s perspective, it simplifies CX: just configure one server.

B. BioThings.io MCP Server. The **BioThings.io MCP Server** (Augmented Nature) connects to the BioThings ecosystem, notably [MyGene.info](#) and [MyVariant.info](#) (^[12] [lobehub.com](#)). These services aggregate gene and variant annotation from multiple sources. The MCP server routes queries to them directly. For example, a call like `mygene_lookup("BRCA1")` or `myvariant_query(alias="rs1234")`. Tools provided include gene ID conversion, variant annotation, etc. This is valuable for bridging genotypic data to functional annotation at scale, similar to UniProt but focused on high-throughput gene lists.

BioThings API covers thousands of genomes via taxonomy and yields structured JSON for each query. An AI assistant can thus translate between different gene IDs and fetch summaries including known disease associations or transcript models. Because BioThings is high-performance, it supports real-time queries on large batches (e.g. thousands of gene IDs).

Discussion (Integrative MCPs): The advantage of servers like Biotools and BioThings is convenience. They reduce engineering overhead by wrapping multiple resources. In practice, a biotech lab could deploy Biotools on their intranet and give all AI agents access, ensuring consistent data versions. Similarly, BioThings MCP can unify how AI handles gene IDs. These are critical when working with heterogeneous datasets.

However, integrative servers also have risks: if one backend API changes or has downtime, it can degrade the entire server. And provenance may be harder: because data flows through a composite, users must trust the builder’s configuration. That said, open-source MCP servers mitigate this by allowing inspection. The listed Biotools and BioThings servers appear to be community projects (MIT licensed).

11. Case Studies and Real-World Examples

To illustrate the impact of MCP servers in biotech, consider the following **hypothetical case studies** (based on realistic scenarios):

- **Case Study 1: Accelerating Literature Synthesis in Drug Discovery**

A pharmaceutical research team is exploring a new target, Protein X. Instead of manually searching PubMed and clinical databases, they deploy an MCP-enabled AI assistant. The team configures the *PubMed MCP server* and the *OpenFDA MCP server*. A researcher asks, "Summarize all recent research on inhibitors of Protein X, and list any related clinical trials." The AI uses `pubmed_search_articles("Protein X AND inhibitor", date_range="2022-2025")` to fetch key papers ([1] www.mcpnow.io), then calls `clinicaltrials_search_studies("Protein X", program="phase 2")` for trials on this protein ([10] github.com). It compiles a report citing specific articles and trial IDs. Because the prompts led to structured data from authoritative sources, the output is backed by evidence. According to Taleb *et al.*, LLMs integrated this way can significantly outpace traditional literature-based discovery methods ([15] www.mdpi.com).

- **Case Study 2: AI-Driven Genomic Analysis**

A clinical genetics lab uses an MCP-based pipeline to help interpret patient variants. An LLM assistant is given a list of observed gene mutations from a sequencing result. It first maps these with the *Ensembl MCP server* (`get_variants(region=...)`) to identify known SNP IDs. Next, it calls `get_variant_consequences` to predict impacts. (beta.mcp.so). It then uses *Gene Ontology MCP* to fetch affected biological processes, and *GTEX MCP* to see if the gene is highly expressed in disease-relevant tissues. Finally, it queries the *PubMed MCP server* for any literature linking these genes to the patient's phenotype. The AI returns a prioritized list of likely pathogenic variants and relevant citations. This workflow closely mirrors described best practices where LLMs can highlight novel insights from literature and data ([15] www.mdpi.com), but now with direct data support.

- **Case Study 3: Automated Biotech Research Assistant**

A startup is developing a novel enzyme. They want to annotate its sequence and find similar enzymes, then gather structural models. They engage an AI with *UniProt MCP*, *AlphaFold MCP*, and *STRING MCP*. The AI queries `search_proteins("enzyme kinase", organism="yeast")`, then filters hits to one sequence. It retrieves the sequence via `get_protein_sequence`. Then it checks AlphaFold with `get_structure("UNIPROT_ID")` to get a 3D model ([40] github.com) and assesses confidence. It next uses STRING (`get_network_interactions`) to find co-expressed partners. Throughout, the AI generates a markdown report including PDB-format files and network graphs. The entire analysis is done without manual database querying.

- **Case Study 4: Clinical Trial Monitoring**

A biotech company is developing Drug Y for Alzheimer's. They set up an AI agent with *ClinicalTrials.gov MCP* and the *OpenFDA MCP*. The AI periodically (monthly) runs `clinicaltrials_search_studies("Drug Y AND Alzheimer", status="recruiting,active")`, tracking new trials. It also calls `search_drug_recalls("Drug Y")` and `search_drug_adverse_events("Drug Y")` in FDA data to monitor safety signals. Results are logged in a dashboard. This automated monitoring (a kind of "AI vigilante") ensures the team stays informed of developments and can adjust trials accordingly.

These examples, while inspired by current research objectives, demonstrate how MCP servers can be chained to perform complex, **evidence-based** analysis. Without MCP, humans would manually navigate multiple APIs; each MCP interface replaces hours of manual work.

12. Implications, Risks, and Future Directions

The adoption of MCP in biotech has far-reaching implications:

- **Productivity and Innovation:** By lowering integration overhead, MCP enables faster prototyping of AI-assisted workflows. Biotech companies can embed LLM tools into daily R&D. For example, toolchains that once required bioinformatics scripting (e.g. variant annotation pipelines) can be accessed via natural language. This may accelerate timelines from discovery to development.

- **Reproducibility and Auditability:** MCP servers encourage *audit trails*. Each tool call is explicit and logged. This aligns with regulatory science: decisions can be traced back to specific database queries. If an AI suggests a hypothesis, one can examine the exact MCP outputs that supported it. In contrast, black-box LLM hallucinations are harder to verify. As TechRadar notes, MCP supports “built-in observability” and audit logs to ensure compliance ([55] www.techradar.com). In biotech (especially clinical settings), such logs are critical.
- **Security and Compliance:** As noted, MCP introduces new security vectors. The *identity fragmentation* problem ([17] www.techradar.com) arises if different servers have separate access controls. Biotech data is often sensitive (patient info, proprietary gene panels), so careful design is needed. Organizations must apply fine-grained entitlements to each MCP server. They should register and vet servers – e.g., only allow known packages (maybe from an internal MCP registry) to connect. The Postmark incident ([18] www.itpro.com) vividly shows that an attacker slipping in a malicious MCP can exfiltrate data. In biotech, where data privacy (HIPAA, GDPR) is paramount, even seemingly innocuous MCP calls (e.g. a query that includes patient identifiers) must be carefully authenticated and encrypted. We expect that **MCP server certification frameworks** will emerge (e.g. the MCPHub certification mentioned by BioMCP ([56] github.com)) to address this. In the interim, best practices include zero-trust (each MCP query token-limited to only necessary scopes) and continuous monitoring.
- **Economic Model and Vendor Participation:** Most current MCP servers are open-source, but we may see commercial offerings. For example, Workato’s enterprise MCP platform ([16] www.axios.com) suggests organizations will pay for secure MCP management. In biotech, cloud providers may spin up official MCP services (e.g. AWS might host a Genes API server). We might also see hybrid models: proprietary clinical databases offering MCP endpoints under license. Pricing/licensing could become an issue: MCP was conceived as open, but proprietary databases (medical records, industrial data) might offer MCP-like APIs behind paywalls. The open standard nature of MCP makes it easier to switch tools, reducing lock-in – a net positive.
- **Standards and Interoperability:** The rapid growth of MCP raises questions of standard schemas and versioning. If everyone implements MCP differently, LLM agents may struggle to use multiple servers consistently. Fortunately, the **MCP specification** (modelcontextprotocol.io) defines a common JSON schema. Many servers here (cyanheads, Augmented Nature, JackKuo, etc.) appear to adhere to common patterns (tools with arguments, returning JSON). Going forward, industry alliances (Anthropic-led working groups) will likely formalize best practices. Pathology wise, we anticipate standardized schemas for common data types (e.g. gene query results, compound query results). This will make building interoperable pipelines easier.
- **AI Capability Shifts:** From a technology perspective, enabling direct database access effectively *lowers the bar* on what LLMs must memorize. GPT may not know every protein sequence, but it can fetch it on demand. This shift means AI progress will be driven less by parameter count and more by ecosystem capability. In bioinformatics, this enables smaller or specialized models to remain useful longer, since heavy data can be offloaded. It also means continuous learning: LLMs can adapt to updated data without retraining, by simply querying the latest entries.

Future Directions: We expect the MCP ecosystem to expand dramatically. Potential upcoming servers include metabolomics (MetaboLights), imaging (DICOM PACS), environmental data (for ecology biotech), and more. Integration with **knowledge graphs** (like Monarch Initiative for phenotype-genotype links) could also be MCP-enabled. On the AI side, agents will become more autonomous, orchestrating chains of MCP calls (multi-hop API reasoning). Tools for federating across servers (MCP proxies, cross-server joins) will evolve.

In biotech R&D, MCP is likely to enable **“Agentic Labs”** where an AI agent runs entire analyses: designing experiments, fetching data, and interpreting results with minimal human scripting. However, the success of this vision hinges on trust and user oversight. LLM hallucination remains a risk if data is misused or mis-interpreted, so firms will likely implement human-in-the-loop steps for critical decisions.

13. Conclusion

The Model Context Protocol promises to be a **game-changer for biotech**. By providing a standardized bridge between AI models and domain-specific data sources, MCP servers unlock capabilities previously unseen. Our survey of the *Top 20 MCP Servers for Biotech* reveals a rich landscape: from core scientific literature (PubMed,

bioRxiv) to genomic resources (Ensembl, NCBI, GTEx), protein databases (UniProt, PDB, AlphaFold), pathways (Reactome), chemical repositories (PubChem, ChEMBL, SureChEMBL, OpenFDA), and clinical data (ClinicalTrials). These servers are already mature enough for real-world use, each offering dozens of "tools" for precise tasks. Used together, they allow LLMs to operate as true scientific assistants – fetching data, analyzing it, and citing credible sources as they reason.

We have provided detailed analysis and citations for each server and category. Specific data points (e.g. number of tools, example queries) and expert commentary have been included to illustrate utility and ground our claims in evidence (^[1] www.mcpnow.io) (^[9] github.com). For instance, an LLM can search PubMed with the same ease as it chats, a scenario made possible only by tools like the PubMed MCP server (^[1] www.mcpnow.io). Case studies demonstrate how integrated workflows could look in drug discovery or genomics, highlighting the **transformative potential** in reducing manual effort and error.

At the same time, we have emphasized **responsible deployment**. Security analyses (^[17] www.techradar.com) (^[18] www.itpro.com) warn that the power of MCP comes with new attack surfaces; biotech firms must design IAM and verification systems accordingly. We noted the broader industry context: enterprise MCP platforms are emerging (^[16] www.axios.com), and LLM benchmarks now often assume tool-calling ability. The future will likely see MCP become an integral part of bioinformatics infrastructure, just as REST APIs are today.

In closing, bridging AI and biotechnology through MCP servers heralds an agile new era of discovery. As one TechRadar analyst summarized, MCP "removes the burden of custom integration" and transforms AI from passive analysis to **active automation** (^[20] www.techradar.com) (^[21] www.techradar.com). For biotech researchers, this means unlocking data-driven insights at unprecedented speed and scale. With careful design and oversight, we expect MCP-enabled AI agents to become indispensable in labs and clinics, turning what are now painstaking manual queries into instant, reliable answers.

Sources: All statements above are drawn from the literature on Model Context Protocol and its applications, including product documentation for the MCP servers, news reports on MCP adoption and risks, and domain references. Key sources include MCP server readmes and registries (^[1] www.mcpnow.io) (^[9] github.com) (^[10] github.com), TechRadar/Axios articles on MCP (^[19] www.axios.com) (^[20] www.techradar.com) (^[17] www.techradar.com), and recent scientific reviews on LLMs in bioinformatics (^[15] www.mdpi.com), among others. Each claim is supported by the cited references.

External Sources

- [1] <https://www.mcpnow.io/en/server/pubmed-cyanheads-pubmed-mcp-server#:~:A%20p...>
- [2] <https://lobehub.com/mcp/jackkuo666-medrxiv-mcp-server#:~:~:~:~:F0%9...>
- [3] <https://lobehub.com/it/mcp/augmented-nature-ncbi-datasets-mcp-server#:~:~:~:~:A%20M...>
- [4] <https://lobehub.com/bg/mcp/augmented-nature-uniprot-mcp-server#:~:~:~:~:~:1...>
- [5] <https://github.com/Augmented-Nature/AlphaFold-MCP-Server#:~:~:~:~:~:Overv...>
- [6] <https://mcpservers.org/servers/Augmented-Nature/Reactome-MCP-Server#:~:~:~:~:~:All%2...>
- [7] <https://mcpservers.org/servers/MCPmed/STRINGmcp#:~:~:~:~:~:Inter...>
- [8] <https://github.com/Augmented-Nature/ChEMBL-MCP-Server#:~:~:~:~:~:Core%...>
- [9] <https://github.com/Augmented-Nature/OpenFDA-MCP-Server#:~:~:~:~:~:Drug%...>

IntuitionLabs - Industry Leadership & Services

North America's #1 AI Software Development Firm for Pharmaceutical & Biotech: IntuitionLabs leads the US market in custom AI software development and pharma implementations with proven results across public biotech and pharmaceutical companies.

Elite Client Portfolio: Trusted by NASDAQ-listed pharmaceutical companies including Scilex Holding Company (SCLX) and leading CROs across North America.

Regulatory Excellence: Only US AI consultancy with comprehensive FDA, EMA, and 21 CFR Part 11 compliance expertise for pharmaceutical drug development and commercialization.

Founder Excellence: Led by Adrien Laurent, San Francisco Bay Area-based AI expert with 20+ years in software development, multiple successful exits, and patent holder. Recognized as one of the top AI experts in the USA.

Custom AI Software Development: Build tailored pharmaceutical AI applications, custom CRMs, chatbots, and ERP systems with advanced analytics and regulatory compliance capabilities.

Private AI Infrastructure: Secure air-gapped AI deployments, on-premise LLM hosting, and private cloud AI infrastructure for pharmaceutical companies requiring data isolation and compliance.

Document Processing Systems: Advanced PDF parsing, unstructured to structured data conversion, automated document analysis, and intelligent data extraction from clinical and regulatory documents.

Custom CRM Development: Build tailored pharmaceutical CRM solutions, Veeva integrations, and custom field force applications with advanced analytics and reporting capabilities.

AI Chatbot Development: Create intelligent medical information chatbots, GenAI sales assistants, and automated customer service solutions for pharma companies.

Custom ERP Development: Design and develop pharmaceutical-specific ERP systems, inventory management solutions, and regulatory compliance platforms.

Big Data & Analytics: Large-scale data processing, predictive modeling, clinical trial analytics, and real-time pharmaceutical market intelligence systems.

Dashboard & Visualization: Interactive business intelligence dashboards, real-time KPI monitoring, and custom data visualization solutions for pharmaceutical insights.

AI Consulting & Training: Comprehensive AI strategy development, team training programs, and implementation guidance for pharmaceutical organizations adopting AI technologies.

Contact founder Adrien Laurent and team at <https://intuitionlabs.ai/contact> for a consultation.

DISCLAIMER

The information contained in this document is provided for educational and informational purposes only. We make no representations or warranties of any kind, express or implied, about the completeness, accuracy, reliability, suitability, or availability of the information contained herein.

Any reliance you place on such information is strictly at your own risk. In no event will IntuitionLabs.ai or its representatives be liable for any loss or damage including without limitation, indirect or consequential loss or damage, or any loss or damage whatsoever arising from the use of information presented in this document.

This document may contain content generated with the assistance of artificial intelligence technologies. AI-generated content may contain errors, omissions, or inaccuracies. Readers are advised to independently verify any critical information before acting upon it.

All product names, logos, brands, trademarks, and registered trademarks mentioned in this document are the property of their respective owners. All company, product, and service names used in this document are for identification purposes only. Use of these names, logos, trademarks, and brands does not imply endorsement by the respective trademark holders.

IntuitionLabs.ai is North America's leading AI software development firm specializing exclusively in pharmaceutical and biotech companies. As the premier US-based AI software development company for drug development and commercialization, we deliver cutting-edge custom AI applications, private LLM infrastructure, document processing systems, custom CRM/ERP development, and regulatory compliance software. Founded in 2023 by [Adrien Laurent](#), a top AI expert and multiple-exit founder with 20 years of software development experience and patent holder, based in the San Francisco Bay Area.

This document does not constitute professional or legal advice. For specific guidance related to your business needs, please consult with appropriate qualified professionals.

© 2025 IntuitionLabs.ai. All rights reserved.