VIBE-QC: an open-source, general-purpose electronic-structure code for molecules and solids developed with generative AI

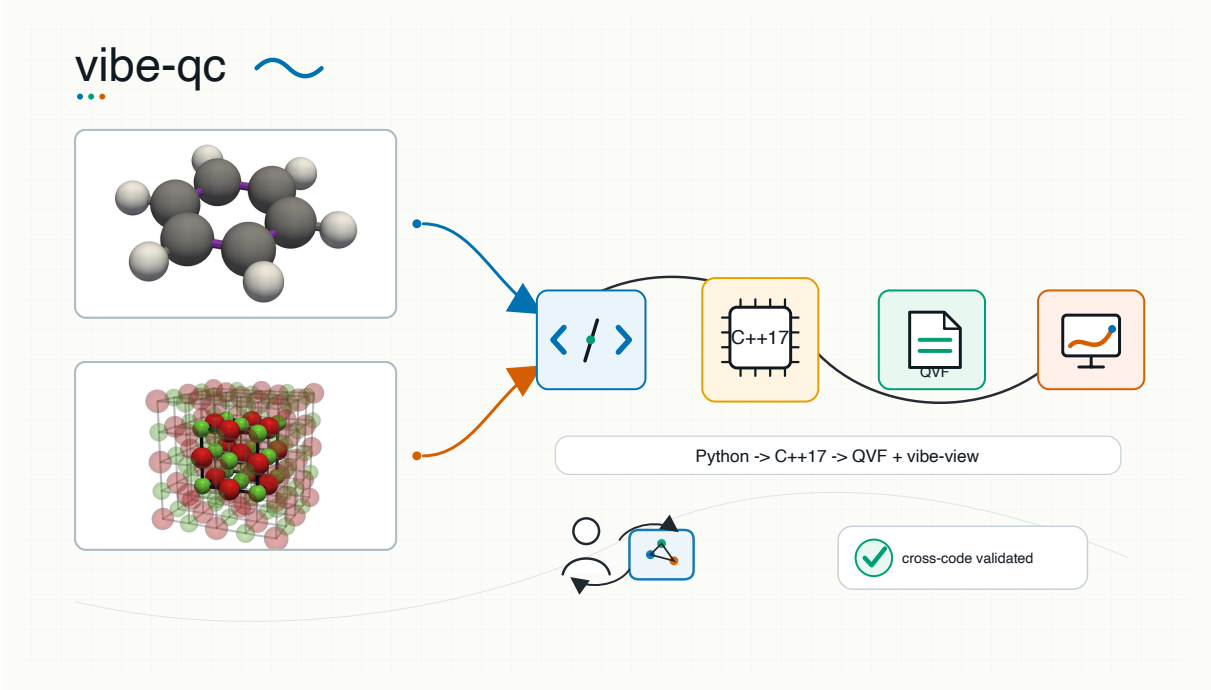
Michael F. Peintinger*

Mulliken Center for Theoretical Chemistry, University of Bonn, Bonn, Germany

Abstract

We present VIBE-QC, a general-purpose open-source electronic-structure code for molecules and solids, developed almost entirely with agentic AI under expert human direction. The implementation comprises a C++17 numerical core and Python calculation drivers. Molecular methods include Hartree–Fock, Kohn–Sham DFT, Møller–Plesset theory and double hybrids, coupled cluster, active-space methods, and excited states. Dispersion corrections, continuum solvation, and effective core potentials are also available. Periodic calculations use Gaussian-basis HF and DFT with Brillouin-zone sampling. Molecular and periodic method coverage differs; the feature matrices specify analytic and numerical derivatives and distinguish validated and experimental implementations. The validation set is a key contribution of this work. It combines independent program comparisons, published benchmarks, and internal consistency tests. Molecular mean-field energies agree at the nanohartree scale, and conventional CCSD(T) energies agree within $3.3 \times 10^{-7}\ E_\mathrm{h}$ for the tested systems. Local-correlation and periodic limitations are reported with the comparisons. Numerical results, calculation metadata, and analysis scripts are supplied as machine-readable Supporting Information. The independently maintained VIBE-VIEW viewer, VIBE-QUEUE scheduler, and QVF toolkit support visualization and job management; TREXIO provides wavefunction interchange.

*Corresponding author. Emails: mpei@vibe-qc.com, mpei@uni-bonn.de



Graphical TOC summary. VIBE-QC is an open-source, general-purpose electronic-structure code for molecules and solids, developed almost entirely with agentic AI under expert human direction. Molecular and periodic calculations use a common interface. Independent reference calculations form a reusable validation set. The VIBE-VIEW viewer, VIBE-QUEUE scheduler, and QVF toolkit are available as separate projects.

Keywords. quantum chemistry, electronic-structure software, periodic systems, generative-AI-assisted development, vibe-coding.

1 Introduction

Implementing and maintaining electronic-structure methods requires substantial work beyond their mathematical formulation. Scientific software often outlives the students and postdoctoral researchers who wrote it, making continuity and reuse important concerns [1, 2]. Community libraries have reduced this burden. Integral engines, exchange–correlation libraries, crystal-symmetry tools, and simulation frameworks let new programs build on established numerical implementations rather than recreate them.

Generative-AI programming assistants allow a domain scientist to direct implementation, testing, debugging, and maintenance of scientific software. Independent reference calculations are required to assess the resulting code.

VIBE-QC began as an experiment in this development model. A first Hartree–Fock prototype was checked against an independent program after one day [3]; the initial broad implementation developed over roughly ten weeks, with validation and revision continuing afterward. Development used conversational programming, often called "vibe-coding", under the direction of one domain scientist. Its public development record [3–6] and the workflow described in Section 6.1 document the division of work. Here, agentic

AI refers to assistants that change code and execute tests under human direction. The author selected the physical models, assessed the results, and approved releases.

The resulting program treats molecules and solids through a common interface and includes numerical kernels, calculation drivers, file interchange, analysis, and documentation. Molecular methods span mean-field theory, perturbation theory, coupled cluster, active-space methods, and excited states. The correlated wavefunction and linear-response methods discussed here are molecular; periodic calculations are assessed at the HF and KS-DFT levels. Molecular ECPs and population analysis are included. The code relies on the reusable libraries described in Section 2.

We compare against ORCA [7], OPENMOLCAS [8, 9], PYSCF [10], CRYSTAL 23 [11], and GPAW [12] because their formulations provide appropriate references for the selected methods. Other established packages include PSI4 [13], TURBOMOLE [14], Q-CHEM [15], and GAUSSIAN [16]. Molecular and periodic coverage in one package is already an established aim, notably in PYSCF and CRYSTAL.

We describe the development and release of this molecular and periodic quantum-chemistry code, its methods, and its architecture. A key contribution is the validation set, which distinguishes agreement between programs under matched settings, accuracy against published benchmarks, and internal consistency. Completed comparisons that exceed the stated tolerance are included. The results and recorded settings are supplied in machine-readable form.

The companion viewer VIBE-VIEW and scheduler VIBE-QUEUE are maintained independently. Sections 2 and 3 describe the methods and interfaces, Sections 4 and 5 report the comparisons and representative calculations, and Section 6 discusses the development method and limitations. Detailed method references, support matrices, convergence data, and implementation notes are in the Supporting Information. The software identities are given in Section 8.

2 Theory and methods

VIBE-QC combines a C++17 numerical core with Python calculation drivers. Gaussian integrals and their derivatives, exchange–correlation functionals, crystal symmetry, ECP integrals, and Fourier transforms use LIBINT [17], LIBXC [18], SPGLIB [19, 20], LIBECPINT [21], and FFTW [22]. The molecular and periodic methods use these libraries for the corresponding numerical operations. SI S10 gives the detailed method-support tables and original references. The support tables identify the systems, settings, and derivatives covered by the comparisons.

2.1 Molecular methods and derivative scope

Molecular calculations cover restricted, unrestricted, and restricted open-shell HF references and KS-DFT. The functional interface supports LDA, GGA, kinetic-energy-density-dependent meta-GGAs, global hybrids, and range-separated hybrids with the VV10 correlation kernel. It does not support Laplacian-dependent meta-GGAs. For open shells, UKS is the default. The conventional restricted open-shell ground-state KS ansatz cannot represent negative spin densities [23]. SI S10.1 explains its optional role and distinguishes modern low-spin excited-state ROKS formulations. Initial guesses and convergence algorithms are also defined and cited there.

Table 1: Scope of the evidence reported in this paper. Detailed spin, derivative, and integral-approximation support is given in SI S10.

Area	Evidence and limitation
Molecular mean field	HF and KS-DFT comparisons against independent programs; DFT agreement depends on quadrature and integral approximations.
Molecular correlation	Small-molecule MP2, CCSD(T), and active-space comparisons; no general production-size memory claim.
Local correlation	Pilot calculations quantify truncation errors relative to canonical results; matching a named PNO setting does not establish cross-code parity.
Periodic methods	Selected GDF and BIPOLE comparisons and convergence ladders; these do not establish accuracy across all cells and meshes.
Derivatives	Analytic and numerical derivatives are identified separately. Energy validation does not validate a gradient or Hessian.

The B3LYP convention is explicit: `b3lyp` and `b3lyp5` select the VWN5 form, corresponding to LIBXC `HYB_GGA_XC_B3LYP5`; `b3lypg` selects the VWN-RPA form. The evaluated identifier is printed in the output. Comparisons must match this choice [24, 25].

Analytic gradients cover HF and the supported semilocal and global-hybrid KS families, with reference-specific restrictions in SI S10. Range-separated hybrid and VV10 gradients are unavailable; optimization uses numerical derivatives of the total energy. Native optimization supplies analytic gradients for a restricted single-state CASSCF subset. The source audit shows that public MP2, CCSD(T), and DLPNO optimization and frequency requests follow the mean-field reference surface; these paths do not provide derivatives of the correlated energy. The water frequencies reported here use finite differences of the analytic RHF gradient through the calculation interface. The derivative support audit distinguishes these numerical Hessians from routines labelled analytic whose response terms still require verification. No general analytic HF/KS Hessian coverage is claimed; in particular, the ASE UKS path uses finite differences. No analytic correlated, meta-GGA, or range-separated Hessian is claimed.

The molecular validation includes MP2 and spin-scaled variants, B2PLYP, conventional and density-fitted CCSD(T), CASCI, CASSCF, NEVPT2, and linear-response excitation energies. D3(BJ) and D4 corrections, continuum solvation, and molecular ECPs extend the tested scope. Their equations, reference implementations, and restrictions are documented with the individual SI comparisons. The local-correlation hierarchy remains a pilot implementation; its measured truncation error is reported separately from canonical cross-code agreement.

2.2 Periodic methods

Periodic HF and KS-DFT use Bloch-summed Gaussian orbitals and explicit Brillouin-zone sampling [26, 27]. The molecular correlated-method coverage does not extend to crystals. The periodic implementation offers Gaussian density fitting (GDF), a direct lattice-sum route termed BIPOLE, and an experimental Gaussian-and-plane-waves (GPW) route with a GAPW augmentation. Only the documented GDF and BIPOLE comparisons support the periodic claims made here.

GDF approximates Coulomb and exchange terms with an auxiliary Gaussian basis [28, 29]. Its accuracy depends separately on the real-space lattice cutoff and the reciprocal-space fitting tail. BIPOLE evaluates the short-range interaction directly and the long-range interaction under an Ewald split [30, 31]. Its

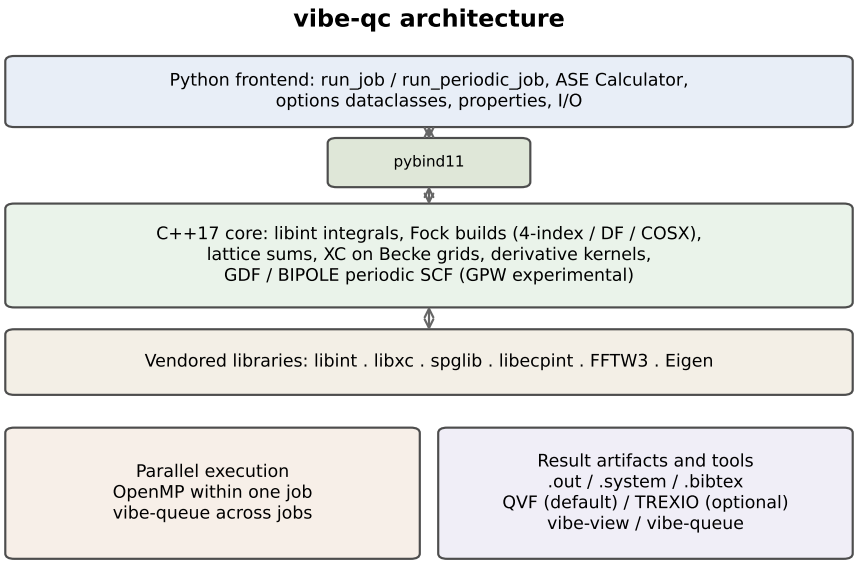


Figure 1: Python calculation drivers over a C++17 numerical core, with established numerical libraries beneath and independently maintained companion projects for visualization, scheduling, and the QVF specification.

production force path uses finite differences of the converged total energy at zero temperature, or Mermin free energy under Fermi smearing; analytic forces remain a research preview. Matching the exchange-divergence convention is essential when comparing periodic absolute energies. SI S4 reports the measured residuals and convergence limits.

Periodic XC quadrature includes cross-cell density contributions and a partition over lattice images, following the Gaussian-basis treatment of Towler, Zupan, and Causà [32]. Symmetry and mesh reduction use SPGLIB. Neither those algorithms nor a converged SCF alone establishes cross-code energy agreement. Variable-cell optimization, periodic ECP calculations through the public drivers, and the experimental GPW/GAPW routes are outside the validated scope.

3 Implementation and companion tools

3.1 Numerical core and calculation interface

The C++17 layer evaluates the integral-driven Fock builds, lattice sums, derivative kernels, and XC grids. It uses EIGEN [33] for dense linear algebra and OpenMP for shared-memory parallelism. Python handles calculation setup, file IO, properties, and workflow control across a `pybind11` interface [34]. The high-level entry points are `run_job` for molecules and `run_periodic_job` for crystals; SI S7 gives input examples. Geometry optimization uses the ASE calculator interface, with a native optimizer available for installations without ASE.

Basis sets can be imported through the JSON interchange format of the Basis Set Exchange [35], as well as Gaussian-94, ORCA, and CRYSTAL text formats. The CRYSTAL format is useful for the pob families [36, 37]; it does not define a distinct ECP capability. Installation and dependency versions are given in Section 8.

3.2 Interchange and the release toolset

VIBE-QC performs the calculations, VIBE-VIEW displays the results, and VIBE-QUEUE (command `vq`) schedules jobs. The three are independently maintained projects. The QVF specification and reference toolkit are maintained separately under Apache 2.0. This article serves as the common release reference for these projects.

Optional wavefunction interchange uses TREXIO [38], with HDF5 and text backends. The writer exports molecular and periodic SCF wavefunctions, scalar ECPs, and molecular CI expansions; a general data API handles additional fields. SI S5 distinguishes automatic export, native reconstruction, and VIBE-VIEW rendering, and lists the corresponding tests and limitations. The molecular interoperability checks passed for the development builds specified there. An actual water TREXIO file was opened and rendered in VIBE-VIEW; its files, build identities, and screenshot are supplied as a supplementary example. These checks are distinct from the publicly tagged release snapshots.

QVF stores visualization and analysis results, calculation metadata, and citations in one archive. VIBE-VIEW reads these archives to display structures, densities, spectra, and periodic properties. Conventional exports also support established tools, including Avogadro [39], IQmol [40], PyMOL [41], and VESTA [42]. The terminal viewer `moltui` by Szenes [43] inspired the terminal visualization feature in VIBE-VIEW. Compatible exports also support the use of `moltui` with VIBE-QC.

VIBE-QUEUE coordinates the calculation campaign. Its scheduler architecture is summarized in SI S6. Calculation inputs and outputs remain usable without this scheduler, and no dependency on it is required to use VIBE-QC. Implementation details are given in the SI and the project documentation.

4 Validation

We assess the code by comparison with independent programs, comparison with published benchmark data, and internal consistency tests. Program comparisons use matched physical and numerical settings. Published benchmarks assess method accuracy. The results of these three types of test are reported separately.

The molecular comparisons use ORCA 6.1.1 and OPENMOLCAS; periodic Gaussian calculations use PYSCF and CRYSTAL 23. Reference programs execute out of process. Inputs must match geometry, basis and ECP, charge and spin, frozen-core treatment, integral approximations, quadrature, convergence settings, and, for crystals, cell and sampling conventions. The method name alone is insufficient to establish equivalent settings. SI S1–S4 documents these settings and the individual comparisons.

4.1 Molecular comparisons

Table 2 summarizes representative results. Molecular HF energies agree at the nanohartree scale. Conventional CCSD(T) agrees within $3.3 \times 10^{-7}\ E_\mathrm{h}$ over eight system–basis combinations, whereas the DF-CCSD comparison on H_2O/cc-pVDZ differs from its conventional reference by $1.4 \times 10^{-4}\ E_\mathrm{h}$. The distinction identifies the integral approximation as part of the comparison, rather than assigning one accuracy to all coupled-cluster routes. CASSCF comparisons require matched active spaces and OPENMOLCAS orbital solutions. Internal FCI and finite-difference checks are labelled separately in the SI.

Table 2: Representative molecular cross-code comparisons. Values are absolute deviations unless stated otherwise. SI S3 gives the individual methods, settings, and recorded release identities.

Method or quantity	System / basis; reference	Agreement
RHF, UHF, ROHF	H_2O, OH, NH_2 / def2-TZVP; ORCA	$1.2 \times 10^{-9}\ E_h$ maximum
KS-DFT ladder	H_2O / def2-TZVP; ORCA	$1.5 \times 10^{-4}\ E_h$ maximum
CCSD(T), conventional	H_2O, CO / eight basis cells; ORCA	$\leq 3.3 \times 10^{-7}\ E_h$
CASSCF	N_2 / cc-pVDZ, (6e,6o); OPENMOLCAS	$\leq 3.0 \times 10^{-10}\ E_h$
TDDFT, TDA	CH_2O / def2-SVP, five roots; ORCA	4.1×10^{-3} eV
Molecular ECP	I_2 / dhf-TZVP; ORCA	$1.32 \times 10^{-8}\ E_h$
S22 interaction energies	22 complexes / PBE-D3(BJ)/def2-SVP; ORCA	0.016 kcal mol^{-1} MAD

Molecular DFT requires particular care. A matched-reference rerun with density fitting disabled in ORCA removed about 92 % of the median deviation in one input family. The remaining difference depends on XC quadrature. Grid ladders on two cases and grid-free HF controls distinguish these contributions, but do not establish an explanation for every functional and system. Those historical inputs also selected a legacy grid and therefore do not measure the shipped default. SI S8 reports the failure groups and the limits of these diagnoses. Disagreements remain visible rather than being removed by a retrospective change of tolerance.

4.2 Published benchmark sets

Published benchmarks assess method accuracy, which is distinct from cross-code parity. All scored entries, including outliers, enter the reported statistics. For QUESTDB [44], the 409 vertical excitation comparisons give mean absolute deviations of 0.21–0.28 eV for the tested adiabatic TDDFT/TDA methods and 0.42 eV for CIS. Doubly excited states and fluorescence entries are excluded on the stated physical and geometry criteria, not by their measured deviation.

Against BH9 [45], the PBE mean absolute deviation for forward barriers is 7.19 kcal mol^{-1}. Reverse barriers give 9.05 kcal mol^{-1}, and reaction energies give 4.93 kcal mol^{-1}. The 951 comparisons quantify PBE's accuracy against the benchmark; they do not by themselves establish implementation parity.

For the S22 subset used by Liakos and co-workers [46], the TightPNO DLPNO-CCSD(T) interaction-energy truncation error relative to the canonical result averages 0.17 kcal mol^{-1} over five complexes, with a maximum of 0.46 kcal mol^{-1}. This is limited pilot evidence. SI S3 gives the full benchmark table, sample sizes by method, exclusions, and the separate electron-propagator accuracy comparison.

4.3 Periodic comparisons and convergence

Periodic agreement depends on both the numerical route and the reference convention. For MgO/STO-3G at Γ, a GDF cutoff ladder reaches $2.7 \times 10^{-5}\ E_h$ against PYSCF with the tested automatic tail setting and $2.2 \times 10^{-7}\ E_h$ with both cutoffs converged. These are archived ladder measurements, rather than scored database comparisons. On LiH the real-space lattice cutoff is the limiting parameter. The two examples do not establish a universal default for all-electron crystals.

The BIPOLE route has an independent diamond/PBE0 check at Γ: orbital energies agree with PYSCF within 0.4 mE_h, the total energy within 1.9 mE_h, and the gap within 9 meV. The database also contains

1 scored periodic cross-code total-energy pass: LiF/pob-TZVP-rev2 RHF on a $2 \times 2 \times 2$ mesh, with the Ewald exchange convention matched between BIPOLE and PySCF, agreeing to $0.25\,\mathrm{m}E_\mathrm{h}$. SI S8 quantifies the reference's fitting uncertainty for this comparison.

Other periodic comparisons retain unresolved cutoff, geometry, or convention differences. In particular, the MgO BIPOLE comparison with Crystal 23 uses slightly different lattice constants and cannot be scored as a same-geometry parity result. SI S4 presents these completed diagnostic calculations with their limitations. GPW/GAPW measurements are not used to support validated release capabilities.

4.4 Regression tests and the validation database

The regression suite tests numerical kernels, interfaces, and failure handling; the independent-program runner tests selected scientific calculations. Test collection counts are not counts of successful cross-code calculations. SI S2 states the snapshot and scope of the test inventory, while SI S9 describes the validation workflow.

The machine-readable database contains 5839 comparisons with 3861 registered case records and 8433 associated program runs. It reports 3329 passes, 1188 missed tolerances, 1073 unscored completed comparisons with stated scientific limitations, and 249 other paired numerical records. Incomplete attempts, superseded comparisons, and results with a disproven molecular identity are omitted from scientific totals; their original records remain recoverable. Completed comparisons with valid identities that miss a tolerance are retained.

Passing rows have different meanings: 1570 compare against an independent program, 1263 belong to benchmark-accuracy sets, 451 compare vibe-qc against itself, and 45 use stored consensus references. Only the first category directly measures agreement with an independent program. Most molecular comparisons use Orca; fewer periodic comparisons are available. SI S8 gives the family and tolerance tables; Section 8 describes the results supplement.

5 Representative calculations

The following calculations illustrate geometry optimization, vibrational analysis, molecular correlation, and effective core potentials. Numerical settings and input examples are given in the SI. The results supplement identifies the corresponding calculations.

5.1 Molecular optimization and vibrational frequencies

The water dimer is optimized at PBE0-D3(BJ)/def2-TZVP through the ASE bridge (script M07), exercising analytic KS gradients and the post-SCF dispersion contribution to the forces under a force-only convergence criterion (maximum force $0.02\,\mathrm{eV\,Å^{-1}}$). The optimized O$\cdots$O separation is 3.0873 Å for the symmetric double-donor arrangement preserved from the starting geometry. The case validates the gradient and dispersion forces on a fixed hydrogen-bond topology and is not a conformational search for the global-minimum dimer structure. Harmonic frequencies and IR intensities for the water monomer (script M14) are obtained from the `run_job` frequency driver, which differentiates the analytic RHF gradient by central finite differences (Section 2.1). The symmetric and antisymmetric stretches and the bend

are reproduced to within 0.11 cm^{-1} of the analytic ORCA Hessian. The associated QVF archives retain the results for inspection in VIBE-VIEW.

5.2 Benzene

Benzene is optimized at PBE0-D3(BJ)/def2-TZVP (script M20). The optimized C–C bond length is 1.3874 Å. Repeating the optimization across the release builds and optimizer generations archived with the paper gives values that spread over 1.8×10^{-4} Å, which is why the value is quoted to four decimals.

5.3 A colocated timing control

Timing is reported only from colocated controls, in which both programs run on the same node with the same allocated CPU count and memory. For hydrogen peroxide at DLPNO-CCSD(T)/cc-pVTZ with one CPU and 6144 MiB each, VIBE-QC required 699.88 s wall time against 156.67 s for ORCA 6.1.1, a factor of 4.47, with peak resident memory of 1015.5 MiB and 2007.2 MiB respectively. VIBE-QC is slower in this comparison; the implementation requires further optimization. A second control on canonical CCSD(T) is given in the Supporting Information.

5.4 Heavy elements through effective core potentials

I_2 at RHF/dhf-TZVP (script M09) exercises the shipped dhf small-core ECP through LIBECPINT on a heavy main-group element. The total energy differs from the matched ORCA calculation by $1.3 \times 10^{-8}\ E_h$, placing the molecular ECP path close to the mean-field parity floor of the all-electron routes.

6 Discussion

6.1 The development methodology

The author selected the physical models, architecture, and validation criteria, reviewed the results, and approved releases. AI assistants translated equations into C++ kernels, implemented the Python calculation interface and tests, and drafted documentation. Anthropic Claude was used for the main development, DeepSeek for routine code generation, and OpenAI tools for debugging and refactoring. Implementations were often reviewed or re-derived in a separate session. SI S9 gives the model versions and development chronology.

Small reference calculations were used during implementation. Agreement with an independent program under matched settings provided a regression target for subsequent optimization and refactoring. Tests also checked symmetry, conservation laws, convergence, and calculation metadata. The test inputs and physical assumptions required review as well as the calculated values.

For example, an apparent MP2 discrepancy arose from different frozen-core conventions. Molecular DFT comparisons required separate assessment of density fitting and quadrature. Periodic comparisons required consistent exchange-divergence corrections and convergence of the lattice and fitting cutoffs. A previously reported XC comparison could not be reproduced and was withdrawn.

The initial implementation was developed over roughly ten weeks, followed by further validation and revision. This observation describes the progress of one project; it is not a controlled estimate of productivity. The work depended on established numerical libraries and human scientific supervision. The validation set is a key contribution: it provides reference comparisons for assessing the code and for subsequent development.

6.2 Design decisions

Separate repositories allow VIBE-VIEW, VIBE-QUEUE, and the QVF toolkit to be used with other calculation programs. TREXIO provides wavefunction interchange, while QVF stores visualization and analysis results. Established workflow systems include QCEngine and QCArchive [47, 48].

The reported calculations use LIBINT for integral evaluation, including derivatives and periodic integrals. Although LIBCINT [49] provides an alternative, no short-term replacement of LIBINT is planned. In the longer term, we aim to replace the current integral mechanism with an approach based on Neese's SHARK integral generation and digestion system [50]. This development is outside the scope of the present work. The reusable trust-region solver `libopentrustregion` [51] and the Molpro iterative-solver library [52] are possible alternatives to the current solvers. They have not been integrated into VIBE-QC.

Mulliken and Löwdin populations remain available for comparison with familiar outputs, but lack a basis-set limit. IAO partial charges [53] are stored in QVF for the supported closed-shell all-electron calculations listed in SI S10; they are not included in the text population reports. Geometry optimization is driven through ASE, with a native optimizer also available. The additional potential-energy-surface drivers are native code rather than ASE workflows. Their scope is described in the software documentation; they are not used as validated demonstrations in this paper.

6.3 Scope of the claims

Molecular HF and ECP comparisons establish tight agreement on the reported cases. DFT and correlated comparisons require explicit grids, auxiliary bases, frozen-core conventions, and numerical thresholds. The MP2, CCSD(T), and active-space evidence is concentrated on small systems. It does not establish production-size memory coverage or accuracy of every spin and derivative combination. Local correlation remains a pilot capability with measured truncation errors.

Periodic tests include GDF cutoff-convergence calculations and BIPOLE comparisons, including the multi-$\boldsymbol{k}$ LiF case. Appropriate cutoffs still need to be established for each crystal. GPW/GAPW, variable-cell optimization, public periodic ECP execution, and correlated periodic methods are outside this paper's validated scope. The same applies to unsupported analytic derivatives and the excited-state gradient whose energy surface has not been verified.

Shared-memory OpenMP is the validated parallel execution mode. Independent calculations can be distributed through the queue. Production MPI execution, GPU compute kernels, and binary wheels are not included in the reported tests. Implementation updates are recorded in the issue tracker and software documentation.

6.4 Reproducibility and provenance

The reported calculations used releases in the 0.15–0.17 series. Each result retains the recorded build identity. The supplementary CSV and JSON tables preserve stable case, run, and comparison identifiers, values, units, comparison outcomes, and available method and runtime metadata. Their scripts reproduce the reported aggregates. Empty historical metadata fields mean unrecorded information, not a default setting. The supplement supports reanalysis of the results; it is not a complete archive for rerunning every original calculation.

Reproduction of a calculation requires the source code, inputs, numerical settings, and software dependencies. The AI conversations used during development are not required to execute the resulting program.

7 Conclusions

We have presented VIBE-QC, a general-purpose open-source quantum-chemistry code for molecules and solids developed almost entirely with agentic AI under expert human direction. The program provides molecular mean-field, correlated, active-space, and excited-state methods, and periodic HF and DFT through a common Python interface and C++17 core. The implementation uses established numerical libraries.

The validation set is a key contribution of this work. Independent program comparisons, published benchmarks, and regression tests assess the implemented methods. The support matrices distinguish molecular and periodic coverage, analytic and numerical derivatives, and experimental implementations. Measured disagreements and limitations are included with the results.

The calculation code, VIBE-VIEW viewer, VIBE-QUEUE scheduler, and QVF toolkit are available as separate open-source projects. TREXIO supports wavefunction interchange. Machine-readable results, calculation metadata, and analysis scripts are supplied with the article for reuse and independent analysis.

8 Software and data availability

Software. VIBE-QC, VIBE-VIEW, and VIBE-QUEUE use the Mozilla Public License, version 2.0; the QVF specification and toolkit use Apache 2.0. Development remains on the project's self-hosted server, where the development and validation tooling is integrated. Source access there was provided on request through deploy keys for server protection. Public GitHub mirrors provide direct anonymous access. An automatic mirroring mechanism updates the corresponding repository whenever a new version of any of the four codebases is released. Their public repositories are `https://github.com/vibe-qc/vibe-qc`, `https://github.com/vibe-qc/vibe-view`, `https://github.com/vibe-qc/vibe-queue`, and `https://github.com/vibe-qc/qvf`. Anonymous source retrieval was verified on 16 September 2026. Documentation is hosted at `https://vibe-qc.com`.

The paper's source baseline is VIBE-QC v0.17.1. Its public GitHub snapshot resolves to `164542af38178d a1c62717524a13ccc824075302`; the original source identity is `b4f6035e40d5f15e239c6c45845a ba5e33f316f0`. The public repository contains sanitized source snapshots rather than identical original

Git history. Historical calculations retain their recorded builds. SI S1 lists the verified companion tags and distinguishes them from the later builds used for the TREXIO demonstration.

Results. The journal Supporting Information includes the numerical results as UTF-8 CSV and newline-delimited JSON, with stable identifiers, units, method and settings metadata, comparison outcomes, a schema, analysis scripts, and SHA-256 checksums. Summary tables and benchmark statistics can be reproduced from this package without access to the calculation archive. A separate TREXIO water example contains the actual exported wavefunction, imported QVF files, numerical checks, build identities, and the recorded VIBE-VIEW rendering. SI S8 explains the results selection and historical metadata limitations. The supplement is a results dataset, not a complete native calculation archive.

Installation. The source installation uses `scripts/install.sh` to build the native dependencies; the source can be retrieved with `git clone https://github.com/vibe-qc/vibe-qc.git`. There is no binary wheel. SI S1 gives the dependency versions, hardware, and comparison protocol. OpenMP is the validated parallel execution mode.

Acknowledgments

VIBE-QC rests on the open-source work of others. The author thanks E. F. Valeev (LIBINT), S. Lehtola and the LIBXC developers, A. Togo (SPGLIB), R. A. Shaw and J. G. Hill (LIBECPINT), the EIGEN and `pybind11` teams, the FFTW developers, A. Hjorth Larsen and the ASE developers, and the TREXIO developers, whose libraries and formats VIBE-QC builds on, and K. Szenes for `moltui`, which inspired the terminal visualization feature in VIBE-VIEW. The author is grateful to T. Bredow at the Mulliken Center for Theoretical Chemistry, University of Bonn, for the pob basis-set work that motivates the periodic calculations, and to the ORCA, OPENMOLCAS, PYSCF, CRYSTAL, and GPAW teams, whose programs serve as the cross-code oracles. The anonymous reviewer of the first version of this paper is thanked for a review that improved the code as well as the text. The code was developed with the assistance of the generative-AI models named in Section 6.1.

Development of VIBE-QC is fully self-financed by the author and received no external, institutional, or commercial funding.

Author contributions

Michael F. Peintinger: conceptualization, methodology, software architecture, validation, formal analysis, and writing. The generative-AI tools described in Section 6.1 (Anthropic Claude Opus 4.7, Opus 4.8, Opus 5, and Fable 5; DeepSeek Flash, Pro, and v4 Pro; and OpenAI Codex 5.5 and ChatGPT 5.6 Sol) assisted code generation, test scaffolding, documentation, and manuscript drafting under the author's scientific control. The author reviewed, verified, and edited the AI-assisted output and takes responsibility for the design, validation, interpretation, and manuscript. No AI tool is listed as an author.

Conflicts of interest

The author declares no conflicts of interest.

Data availability

The machine-readable results and TREXIO rendering example accompany this article as journal Supporting Information. The dataset includes the reported values, recorded settings, analysis scripts, schema, and checksums; its scope and the software identities are described in Section 8.

References

[1] S. Lehtola and A. J. Karttunen. "Free and open source software for computational chemistry education". In: *WIREs Computational Molecular Science* 12.5 (2022), e1610. DOI: 10.1002/wcms.1610.

[2] S. Lehtola. "A call to arms: Making the case for more reusable libraries". In: *The Journal of Chemical Physics* 159.18 (2023), p. 180901. DOI: 10.1063/5.0175165.

[3] M. F. Peintinger. *I vibe-coded a quantum chemical program in one day*. https://peintinger.com/2026/04/22/i-vibe-coded-a-quantum-chemical-program-in-one-day/. Apr. 2026.

[4] M. F. Peintinger. *The giants vibe-qc stands on: a guided tour of the stack*. https://peintinger.com/2026/04/23/the-giants-vibe-qc-stands-on-a-guided-tour-of-the-stack/. Apr. 2026.

[5] M. F. Peintinger. *vibe-qc documentation as educational artefact*. https://peintinger.com/2026/04/26/vibe-qc-documentation-as-educational-artefact/. Apr. 2026.

[6] M. F. Peintinger. *Using Claude to drive vibe-qc*. https://vibe-qc.com/using_claude.html. Part of the vibe-qc documentation, release branch. 2026.

[7] F. Neese. "Software update: the ORCA program system—version 6.0". In: *WIREs Computational Molecular Science* 15.1 (2025), e70019. DOI: 10.1002/wcms.70019.

[8] I. Fdez. Galván, M. Vacher, A. Alavi, C. Angeli, F. Aquilante, J. Autschbach, J. J. Bao, S. I. Bokarev, N. A. Bogdanov, R. K. Carlson, et al. "OpenMolcas: from source code to insight". In: *Journal of Chemical Theory and Computation* 15.11 (2019), pp. 5925–5964. DOI: 10.1021/acs.jctc.9b00532.

[9] G. Li Manni, I. F. Galván, A. Alavi, F. Aleotti, F. Aquilante, J. Autschbach, D. Avagliano, A. Baiardi, J. J. Bao, S. Battaglia, et al. "The OpenMolcas web: a community-driven approach to advancing computational chemistry". In: *Journal of Chemical Theory and Computation* 19.20 (2023), pp. 6933–6991. DOI: 10.1021/acs.jctc.3c00182.

[10] Q. Sun, X. Zhang, S. Banerjee, P. Bao, M. Barbry, N. S. Blunt, N. A. Bogdanov, G. H. Booth, J. Chen, Z.-H. Cui, et al. "Recent developments in the PySCF program package". In: *The Journal of Chemical Physics* 153.2 (2020), p. 024109. DOI: 10.1063/5.0006074.

[11] A. Erba, J. K. Desmarais, S. Casassa, B. Civalleri, L. Donà, I. J. Bush, B. Searle, L. Maschio, L. Edith-Daga, A. Cossard, C. Ribaldone, E. Ascrizzi, N. L. Marana, J.-P. Flament, and B. Kirtman. "CRYSTAL23: a program for computational solid state physics and chemistry". In: *Journal of*

Chemical Theory and Computation 19.20 (2023), pp. 6891–6932. DOI: 10.1021/acs.jctc.2c00958.

[12] J. Enkovaara, C. Rostgaard, J. J. Mortensen, J. Chen, M. Dułak, L. Ferrighi, J. Gavnholt, C. Glinsvad, V. Haikola, H. A. Hansen, H. H. Kristoffersen, M. Kuisma, A. H. Larsen, L. Lehtovaara, M. Ljungberg, O. Lopez-Acevedo, P. G. Moses, J. Ojanen, T. Olsen, V. Petzold, N. A. Romero, J. Stausholm-Møller, M. Strange, G. A. Tritsaris, M. Vanin, M. Walter, B. Hammer, H. Häkkinen, G. K. H. Madsen, R. M. Nieminen, J. K. Nørskov, M. Puska, T. T. Rantala, J. Schiøtz, K. S. Thygesen, and K. W. Jacobsen. "Electronic structure calculations with GPAW: a real-space implementation of the projector augmented-wave method". In: *Journal of Physics: Condensed Matter* 22.25 (2010), p. 253202. DOI: 10.1088/0953-8984/22/25/253202.

[13] D. G. A. Smith, L. A. Burns, A. C. Simmonett, R. M. Parrish, M. C. Schieber, R. Galvelis, P. Kraus, H. Kruse, R. Di Remigio, A. Alenaizan, et al. "Psi4 1.4: Open-source software for high-throughput quantum chemistry". In: *The Journal of Chemical Physics* 152.18 (2020), p. 184108. DOI: 10.1063/5.0006002.

[14] S. G. Balasubramani, G. P. Chen, S. Coriani, M. Diedenhofen, M. S. Frank, Y. J. Franzke, F. Furche, R. Grotjahn, M. E. Harding, C. Hättig, et al. "TURBOMOLE: Modular program suite for ab initio quantum-chemical and condensed-matter simulations". In: *The Journal of Chemical Physics* 152.18 (2020), p. 184107. DOI: 10.1063/5.0004635.

[15] E. Epifanovsky, A. T. B. Gilbert, X. Feng, J. Lee, Y. Mao, N. Mardirossian, P. Pokhilko, A. F. White, M. P. Coons, A. L. Dempwolff, et al. "Software for the frontiers of quantum chemistry: An overview of developments in the Q-Chem 5 package". In: *The Journal of Chemical Physics* 155.8 (2021), p. 084801. DOI: 10.1063/5.0055522.

[16] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, G. A. Petersson, H. Nakatsuji, et al. *Gaussian 16, Revision C.01*. Gaussian, Inc., Wallingford, CT, 2016.

[17] E. F. Valeev. *Libint: a library for the evaluation of molecular integrals of many-body operators over Gaussian functions*. https://github.com/evaleev/libint. Version 2.13.1. 2025.

[18] S. Lehtola, C. Steigemann, M. J. T. Oliveira, and M. A. L. Marques. "Recent developments in libxc — a comprehensive library of functionals for density functional theory". In: *SoftwareX* 7 (2018), pp. 1–5. DOI: 10.1016/j.softx.2017.11.002.

[19] A. Togo and I. Tanaka. "Spglib: a software library for crystal symmetry search". In: *arXiv preprint* (2018). arXiv: 1808.01590.

[20] A. Togo, K. Shinohara, and I. Tanaka. "Spglib: a software library for crystal symmetry search". In: *Science and Technology of Advanced Materials: Methods* 4.1 (2024), p. 2384822. DOI: 10.1080/27660400.2024.2384822.

[21] R. A. Shaw and J. G. Hill. "Prescreening and efficiency in the evaluation of integrals over ab initio effective core potentials". In: *The Journal of Chemical Physics* 147.7 (2017), p. 074108. DOI: 10.1063/1.4986887.

[22] M. Frigo and S. G. Johnson. "The design and implementation of FFTW3". In: *Proceedings of the IEEE* 93.2 (2005), pp. 216–231. DOI: 10.1109/JPROC.2004.840301.

[23] J. A. Pople, P. M. W. Gill, and N. C. Handy. "Spin-unrestricted character of Kohn–Sham orbitals for open-shell systems". In: *International Journal of Quantum Chemistry* 56.4 (1995), pp. 303–305. DOI: 10.1002/qua.560560414.

[24] R. H. Hertwig and W. Koch. "On the parameterization of the local correlation functional. What is B3LYP?" In: *Chemical Physics Letters* 268.5–6 (1997), pp. 345–351. DOI: 10.1016/S0009-2614(97)00207-8.

[25] S. Lehtola and M. A. L. Marques. "Reproducibility of density functional approximations: How new functionals should be reported". In: *The Journal of Chemical Physics* 159.11 (2023), p. 114116. DOI: 10.1063/5.0167763.

[26] C. Pisani, R. Dovesi, and C. Roetti. *Hartree–Fock Ab Initio Treatment of Crystalline Systems*. Vol. 48. Lecture Notes in Chemistry. Berlin, Heidelberg: Springer, 1988. DOI: 10.1007/978-3-642-93385-1.

[27] H. J. Monkhorst and J. D. Pack. "Special points for Brillouin-zone integrations". In: *Physical Review B* 13.12 (1976), pp. 5188–5192. DOI: 10.1103/PhysRevB.13.5188.

[28] H.-Z. Ye and T. C. Berkelbach. "Fast periodic Gaussian density fitting by range separation". In: *J. Chem. Phys.* 154.13 (2021), p. 131104. DOI: 10.1063/5.0046617.

[29] Q. Sun, T. C. Berkelbach, J. D. McClain, and G. K.-L. Chan. "Gaussian and plane-wave mixed density fitting for periodic systems". In: *The Journal of Chemical Physics* 147.16 (2017), p. 164119. DOI: 10.1063/1.4998644.

[30] P. P. Ewald. "Die Berechnung optischer und elektrostatischer Gitterpotentiale". In: *Annalen der Physik* 369.3 (1921), pp. 253–287. DOI: 10.1002/andp.19213690304.

[31] V. R. Saunders, C. Freyria-Fava, R. Dovesi, L. Salasco, and C. Roetti. "On the electrostatic potential in crystalline systems where the charge density is expanded in Gaussian functions". In: *Molecular Physics* 77 (1992), pp. 629–665. DOI: 10.1080/00268979200102671.

[32] M. D. Towler, A. Zupan, and M. Causà. "Density functional theory in periodic systems using local Gaussian basis sets". In: *Computer Physics Communications* 98.1–2 (1996), pp. 181–205. DOI: 10.1016/0010-4655(96)00078-1.

[33] G. Guennebaud, B. Jacob, et al. *Eigen v3*. https://eigen.tuxfamily.org. 2010.

[34] W. Jakob, J. Rhinelander, and D. Moldovan. *pybind11 — seamless operability between C++11 and Python*. https://github.com/pybind/pybind11. 2017.

[35] B. P. Pritchard, D. Altarawy, B. Didier, T. D. Gibson, and T. L. Windus. "New Basis Set Exchange: An Open, Up-to-Date Resource for the Molecular Sciences Community". In: *Journal of Chemical Information and Modeling* 59.11 (2019), pp. 4814–4820. DOI: 10.1021/acs.jcim.9b00725.

[36] M. F. Peintinger, D. Vilela Oliveira, and T. Bredow. "Consistent Gaussian basis sets of triple-zeta valence with polarization quality for solid-state calculations". In: *Journal of Computational Chemistry* 34.6 (2013), pp. 451–459. DOI: 10.1002/jcc.23153.

[37] D. Vilela Oliveira, J. Laun, M. F. Peintinger, and T. Bredow. "BSSE-correction scheme for consistent Gaussian basis sets of double- and triple-zeta valence with polarization quality for solid-state calculations". In: *Journal of Computational Chemistry* 40.27 (2019), pp. 2364–2376. DOI: 10.1002/jcc.26013.

[38] E. Posenitskiy, V. G. Chilkuri, A. Ammar, M. Hapka, K. Pernal, R. Shinde, E. J. Landinez Borda, C. Filippi, K. Nakano, O. Kohulák, S. Sorella, P. de Oliveira Castro, W. Jalby, P. L. Ríos, A. Alavi,

and A. Scemama. “TREXIO: A file format and library for quantum chemistry”. In: *The Journal of Chemical Physics* 158.17 (2023), p. 174801. DOI: 10.1063/5.0148161.

[39] M. D. Hanwell, D. E. Curtis, D. C. Lonie, T. Vandermeersch, E. Zurek, and G. R. Hutchison. “Avogadro: an advanced semantic chemical editor, visualization, and analysis platform”. In: *Journal of Cheminformatics* 4 (2012), p. 17. DOI: 10.1186/1758-2946-4-17.

[40] A. T. B. Gilbert. *IQmol molecular viewer*. 2024.

[41] Schrödinger, LLC. *The PyMOL Molecular Graphics System*. 2024.

[42] K. Momma and F. Izumi. “VESTA 3 for three-dimensional visualization of crystal, volumetric and morphology data”. In: *Journal of Applied Crystallography* 44.6 (2011), pp. 1272–1276. DOI: 10.1107/S0021889811038970.

[43] K. Szenes. *moltui: a terminal user interface for viewing molecules and crystals*. 2026.

[44] M. Véril, A. Scemama, M. Caffarel, F. Lipparini, M. Boggio-Pasqua, D. Jacquemin, and P.-F. Loos. “QUESTDB: A database of highly accurate excitation energies for the electronic structure community”. In: *WIREs Comput. Mol. Sci.* 11.5 (2021), e1517. DOI: 10.1002/wcms.1517.

[45] V. K. Prasad, A. Otero-de-la-Roza, and G. A. DiLabio. “BH9, a New Comprehensive Benchmark Data Set for Barrier Heights and Reaction Energies: Assessment of Density Functional Approximations and Basis Set Incompleteness Potentials”. In: *J. Chem. Theory Comput.* 18.1 (2021), pp. 151–166. DOI: 10.1021/acs.jctc.1c00694.

[46] D. G. Liakos, Y. Guo, and F. Neese. “Comprehensive Benchmark Results for the Domain Based Local Pair Natural Orbital Coupled Cluster Method (DLPNO-CCSD(T)) for Closed- and Open-Shell Systems”. In: *J. Phys. Chem. A* 124.1 (2019), pp. 90–100. DOI: 10.1021/acs.jpca.9b05734.

[47] D. G. A. Smith, A. T. Lolinco, Z. L. Glick, J. Lee, A. Alenaizan, T. A. Barnes, C. H. Borca, R. Di Remigio, D. L. Dotson, S. Ehlert, et al. “Quantum Chemistry Common Driver and Databases (QCDB) and Quantum Chemistry Engine (QCEngine): Automation and interoperability among computational chemistry programs”. In: *The Journal of Chemical Physics* 155.20 (2021), p. 204801. DOI: 10.1063/5.0059356.

[48] D. G. A. Smith, D. Altarawy, L. A. Burns, M. Welborn, L. N. Naden, L. Ward, S. Ellis, B. P. Pritchard, and T. D. Crawford. “The MolSSI QCArchive project: An open-source platform to compute, organize, and share quantum chemistry data”. In: *WIREs Computational Molecular Science* 11.2 (2020), e1491. DOI: 10.1002/wcms.1491.

[49] Q. Sun. “Libcint: An efficient general integral library for Gaussian basis functions”. In: *Journal of Computational Chemistry* 36.22 (2015), pp. 1664–1671. DOI: 10.1002/jcc.23981.

[50] F. Neese. “The SHARK integral generation and digestion system”. In: *Journal of Computational Chemistry* 44.3 (2023), pp. 381–396. DOI: 10.1002/jcc.26942.

[51] J. Greiner, I.-M. Høyvik, S. Lehtola, and J. J. Eriksen. “A reusable library for second-order orbital optimization using the trust region method”. In: *Journal of Chemical Theory and Computation* 22 (2026), pp. 881–895. DOI: 10.1021/acs.jctc.5c01576.

[52] P. J. Knowles et al. *iterative-solver: a library of iterative solvers for linear and non-linear eigenvalue and linear equations*. 2026.

[53] G. Knizia. “Intrinsic Atomic Orbitals: An Unbiased Bridge between Quantum Theory and Chemical Concepts”. In: *Journal of Chemical Theory and Computation* 9.11 (2013), pp. 4834–4843. DOI: 10.1021/ct400687b.