Changelog

v. 1.06 2020-12-20

Added

- Zz pseudoelement atoms both in polymeric and general-placeholder context
- Option *NPZZ* to allow non-polymer *Zz* atoms (polymeric ones are allowed by default by *Polymers* switch)
- Option *Polymers105* to emulate v. 1.05 treatment of polymers
- Support of simultaneous presence of both source-based and structure-based subunits in copolymer unit (per request of Gerd Blanke, StructurePendium)
- Option *NoFrameShift* (disables frame shift for all CRU's), works for both inchi-1 and API
- In-CRU inner repeats are performed if the switch *FoldCRU* is specified (so *-(CH2CH2)n-* is converted to *-(CH2)n-* and so on)
- Switch *NoEDits* disables both frame shift and CRU folding
- Option *SAtZz* enables stereo at atoms connected to Zz pseudo atoms (default: disabled/ignored)
- Switch *LooseTSACheck* relaxes strictness of tetrahedral stereo ambiguity check for stereo atoms in cycles (useful for dealing with large cycles 'cleaned' by some software), per request from FDA.
- Switch *WMnumber* for InChI Library/inchi-1 (sets timeout in milliseconds, strictly requires long int number), per request from CDK folks
- *NoWarnings* option to inchi-1 and InChI Library: suppress warnings but keep error messages; useful for the long runs on multi-million record inputs
- Switch *MergeHash*:   make combined InChIKey+extra hash(es) if present (inchi-1)
- Switch *PERTHREAD:n* allowing one to process n SDF records in each of mol2inchi threads.
- Use *Tab* as synonym to *Tabbed*, works for inchi-1
- API call IXA_INCHIBUILDER_SetOption_Timeout_Milliseconds()
- IXA_MOL_GetBondOtherAtom() exposed in the public API, as Paul Thiessen suggested
- API call IXA_MOL_ReserveSpace() (necessary in new mode IXA_USES_SMART_ALLOCS)
- More IXA API functionality for polymers (API calls IXA_MOL_CreatePolymerUnit (), IXA_MOL_SetPolymerUnit (), IXA_MOL_GetPolymerUnitId(), IXA_MOL_GetPolymerUnitIndex())
- Made /inchi2inchi calc mode to treat InChI=1//
- Provided a simplistic example of multi-threaded boss-worker InChI generation into mol2inchi demo program (Linux pthreads/Windows threads are used)
- winchi-1: added SDF navigation by keyboard Ctrl-Lt, Ctrl-Rt arrows; Ctrl-G for Goto Compound #

Changed

- Polymers with undefined ("star", or *Zz*) end-groups are represented with explicitly shown in InChI string Zz atoms. If CRU frame shift took place, bonding is changed accordingly
- In InChI part for any polymer CRU , the crossing bond printed first is now that pointing to more senior CRU end ("head")
- Returned value of IXA_INCHIBUILDER_OPTION_LargeMolecules to 11 as it was in v1.05 but 31 in v1.06pre3 (per request by Daniel Lowe)
- Improved performance of IXA memory allocations by using expandable arrays (per request by Daniel Lowe; see #ifdef IXA_USES_SMART_ALLOCS)

- Changed inchi-1 SDF ID value type to unsigned long to account for current CAS numbers which may now be > LONG MAX (thanks to DT for noticing)
- Made return codes of GetInChIFrom...() functions in case of InChI read error corresponding to what
is declared in API description
- Changed INFINITY defined in chi_can2.c to INCHI_CANON_INFINITY to avoid collision with MS' INFINITY defined in VS 2015 math.h (pointed out by DT)
- Changed label of winchi-1 toolbar button "Write result" to more descriptive "Batch process all"
- Changed winchi-1 disk output behavior to avoid unnecessary writing, per ACD/Labs request . Now the program writes log/output/problem files disk only in batch mode, if not otherwise requested
- Now winchi-1 starts in maximized window.
- Made numerous small changes related to the refactoring of code

Removed

- Removed function char base26_checksum( ) (as there is no check character in InChIKey anymore)
- Removed *InChI2InChI* test option of inchi-1 executable (use *RTrip* option of test_ixa demo app instead, or use API directly)

Fixed (security)

- Fixed 22 possibly security-compromising issues found by Google-AutoFuzz of 2018-2019 (reported via SourceForge)
mostly issues related to inadequate handling of invalid/artificial input on inchi2struct conversion that may result in memory corruption, etc.
- Fixed 102 more, then 1 more, possibly security-compromising issues found by Google-AutoFuzz of 2019 (reported by Ian Wetherbee), mostly issues related to inadequate handling of invalid/artificial input on inchi2struct conversion that may result in memory corruption, etc.; some are similar to/same as GAF22
- Fixed 19 more possibly security-compromising issues of 2020 found by google/oss-fuzz
- Fixed 4 possibly security-compromising issues of 2019 (reported by Cure53) issues found with AddressSanitizer, mostly related to inadequate handling of invalid/artificial input that may result in memory corruption, etc.

Fixed (other)

- Fixed normalization bug resulting in change of InChI string upon atom renumbering for some molecules having acidic hydroxy group at cationic heteroatom center. Total of ~1700 PubChem Compound entries out of ~102M are favorably affected, see KNOWNISSUES.md section on 1.05 issues. In particular, this fix repairs the issue reported by Lutz Weber, Ontochem, on 2020-03-03.
- Fixed bug resulting in InChI Error -30010 (STEREOCOUNT_ERR) -- which appeared on CIDs 124897603, 124921144 as was found on massive testing with PubChem dataset. "The failure occurs when one of two or more constitutionally equivalent undefined stereocenters has been removed due to stereo equivalence of its two attachments." Thanks, DT!
- Fixed bug reported by Andrew Dalke and separately by Burt Leland: H isotope with insane mass difference has been silently consumed which sometimes resulted in memory corruption
- Fixed bug of not recognizing spiro chirality like in olean, described in: Maeda et al., Biophys.Physicobiol., 2018, 87 (thanks to DT)

- Added check to fix bug with NULL szXtra in ikey_dll.c, thanks to Wolf-Dietrich Ihlenfeldt
- Fixed "data race" bug(s) reported by John Salmon in inchi-discuss and similar ones (by adding el_number constants instead of static variables)
- Fixed critical race condition bug pointed out by Karl Nedwed, strutil.c
- Fixed bug in polymer CRU canonicalization in inchi2struct mode at comparing seniority of 'junior members' for various possible pairs of 'star' atom attachment points
- Fixed bug in inchi2struct for polymers exhibited at multipliers in the formula
- Fixed issue in polymer treatment: enabled CRU frame (phase) shift only for "head-to-tail" connection; fixed several bugs for star-ended CRU's sometimes resulting in crash
- Fixed bug which caused crash on read V3000 file alkane-c4000.mol
- Fixed bug "for V3000 AuxInfo misses coordinates /rC:" (tested: methane.sdf, pdb-1000-to-1023-atoms.sdf).
- Fixed several bugs in AuxInfo output (some of general nature and some specifically of IXA)
- Fixed the case of erroneous behavior of inchi-1 with "/D" option
- Added treating of mistakenly overlooked case of zero bond number in MOL V3000 parser (thanks to Burt Leland)
- Fixed typo in ichirvr2.c#2838 and removed junk UTF symbols appeared due to copy-pasting from PDF
- Fixed minor issues with output in /Tabbed mode
- Fixed: multi-threaded mol2inchi did not treat MOL files (SDF was OK)
- Fixed issue of winchi-1 making no output to result's text window after calling "Process all" from record !=1
- Fixed issue of unnecessarily repeating some output lines in winchi-1 InChI/AuxInfo/annotation window

v. 1.05 release / 2017-02-04