

ChemSpectra: A Web-based Spectra Editor for Analytical data

Yu-Chieh Huang

Karlsruher Institut für Technologie <https://orcid.org/0000-0002-4261-9886>

Pierre Tremouilhac

Karlsruher Institut für Technologie <https://orcid.org/0000-0002-0487-3947>

An Nguyen

Karlsruher Institut für Technologie <https://orcid.org/0000-0002-1692-6778>

Nicole Jung (✉ nicole.jung@kit.edu)

Karlsruher Institut für Technologie <https://orcid.org/0000-0001-9513-2468>

Stefan Bräse

Karlsruher Institut für Technologie <https://orcid.org/0000-0003-4845-3191>

Software

Keywords: Spectroscopy, analysis, NMR, IR, mass spectrometry, JCAMP-DX

Posted Date: July 21st, 2020

DOI: <https://doi.org/10.21203/rs.3.rs-44215/v1>

License:  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Version of Record: A version of this preprint was published on February 10th, 2021. See the published version at <https://doi.org/10.1186/s13321-020-00481-0>.

Abstract

ChemSpectra, a web-based software to visualize and analyze spectroscopic data, integrating solutions for IR (infrared spectroscopy), MS (mass spectrometry), and one dimensional ^1H and ^{13}C NMR (proton and carbon nuclear magnetic resonance) spectroscopy, is described. ChemSpectra serves as work instrument for the analysis of the most often used types of one dimensional spectroscopic data in synthetic (organic) chemistry research. The software is able to deal with the open file formats JCAMP-DX (IR, MS, NMR) and mzML (MS), allowing the further extension to other formats as exemplarily shown for selected proprietary mass spectrometry data files of type RAW and NMR spectra files of type FID. The JavaScript based editor can be integrated with other software, which was shown for the *Chemotion* electronic lab notebook (ELN) and *Chemotion* repository, demonstrating the implementation to a digital work environment that offers additional functionality and sustainable research data management options. ChemSpectra supports different functions for the work with spectroscopic data such as zoom functions, peak picking and the automatic peak detection according to a default or manually defined threshold. NMR specific functions include the definition of a reference signal, the integration of signals, coupling constant calculation and multiplicity assignment. Embedded to a web application such as an ELN or a repository, the editor additionally allows the association to a sample and a file management. The file management supports the storage of the original spectra along with the last edited version and an automatically generated image of the spectra in png format. To maximize the benefit of the spectra editor for e.g. ELN users, an automated procedure for the transfer of the detected or manually chosen signals to the ELN was implemented. ChemSpectra is released under the AGPL license to encourage its re-use and further developments by the community.

Background

The online and interactive visualization of spectroscopic data is crucial for modern scientific work to be able to evaluate scientific data and to analyze it. Web-based solutions are beneficial because of their platform-independent use and few system requirements. As web-based software developments usually can be embedded to manifold projects, they may serve as valuable contribution to existing databases and information systems (like electronic lab notebooks, ELNs, or repositories). In chemistry research, in particular the information from ^1H and ^{13}C NMR, IR, and mass spectroscopic experiments is of high importance as these four techniques are essential for the identification of molecules. Due to their significance, they generally form the standard set of analytical data that has to be provided along with the scientific publication of synthetic results. NMR, MS and IR data can be analyzed either manually from printed spectra or they can be analyzed in detail using commercial software or free stand-alone tools. The available commercial or non-open source software products usually include manifold functionalities for processing, visualization, editing and documentation but they have to be installed and work in most of the cases in a non-embedded manner. Examples for such software options are MestreNova, ChemAxon, TopSpin (NMR) or Spectragryph (IR). Besides professional tools as the given ones, only a few web-based developments are available as an Open Source. For the spectra types NMR, MS and IR which are

considered in this work, the web-based visualization tools JSpecView, NMRPro, MetaboAnalyst, MetaboHunter, COLMAR, jsNMR, and SpeckTackle are known. Some of them are already integrated in web services, e.g. SpeckTackle is used in MetaboLights for NMR/MS data. Other databases such as the databases Human Metabolome Database HMDB or DrugBank are supported by additional editors that are developed internally explicitly for those databases.

Due to the need for advanced spectra editors for the visualization but also for the analysis and editing of spectroscopic data such as NMR, MS and IR data with peak-picking, NMR signal integration, coupling constant calculation and multiplicity assignment, we initiated a project that is based on currently available source code and tools from own developments. The aim of the development is the extension of the applicability of web-based editors to enable their use for enhanced data management tasks in particular for web based data management systems such as ELNs and repositories.

Implementation

ChemSpectra consists of three modules, the react-spectra-editor, chem-spectra-client, and chem-spectra-app. For the client-side implementations (react-spectra-editor and chem-spectra-client) the software is written in the programming language JavaScript using the framework React.js which offers a user-friendly single page application to upload, read, edit, and download spectra. JavaScript was chosen due to its benefits for interactive web tools in order to facilitate the embedding of the developments to other web-based software. The jcampconverter library inside the react-spectra-editor extracts spectroscopic data of the composed JCAMP-DX file which is sent from the server side.

The server-side spectra handling (chem-spectra-app) is based on python and was built using a modified version of NMRglue and SciPy. Python as backend programming language on the framework Flask provides the data processing and ensures the compatibility with and re-use of the previously developed systems.

ChemSpectra can be used as a stand-alone software to be offered as an independent web service or can be used for other web developments. The stand-alone application supports the visualization and analysis of spectra, but its functions are limited in comparison to an embedded version, as the information stored in the browser exists temporary and is not persisted permanently. To show the advantages of the ChemSpectra development as embedded application offering the full available functionality, it was incorporated to the web applications *Chemotion* ELN and repository, which are developments of our research group reported earlier. Examples for further functions are the storage and management of the files that were edited by ChemSpectra. Figure 1 gives an overview of the main parts of the ChemSpectra software and the implemented processes for the stand-alone version (green arrows) and the exemplarily implementation with the *Chemotion* applications (workflow given in blue). While, for the stand-alone implementation, the chem-spectra-app communicates directly with the chem-spectra-client, the communication for the embedded software is managed by the server of the ELN or repository. If ChemSpectra is embedded into other web applications, further systems and work processes such as a

data provider can be added to the overall workflow. In the herein depicted example, an instrument server that provides analytical data that were collected by a data collector of the ELN is connected.

ChemSpectra has been optimized and was tested thoroughly on the Browser Chrome. A stand-alone server version and a version of ChemSpectra embedded to the *Chemotion* ELN and repository are available for demonstration at the *Chemotion* project website, www.chemotion.net. In addition, the source code of the project for chem-spectra-app, chem-spectra-client, and react-spectra-editor can be retrieved from github. The source code is released as an Open Source under the license AGPL version 3.

Results

The main part of the ChemSpectra software is the react-spectra-editor which displays the three types NMR, MS and IR data. The type of spectra is extracted automatically from the provided files. Depending on the extracted type, one of two layouts available for visualization is used: the line plot (NMR and IR data) or the bar graph (MS data). To edit the provided data with ChemSpectra, a control panel offers generic and data-type specific actions to analyze and configure the given data (Figs. 2 and 3). The generic actions are available for NMR, IR and MS data and allow to (1) zoom in and out (2) adjust the threshold that is given as default for each spectra type, and (3) extract the peaks and write them in a list form. With respect to the selected signals in the spectra, the user can select the amount of displayed digits for each signal and in which order the signals should be summarized (descending or ascending).

The backend part of the ChemSpectra software, the chem-spectra-app, manages the decoding and composing of spectra files, the peak-picking, and the image generation as basic parts for the transformation of the given spectra. Currently, the chem-spectra-app accepts the file extensions jcamp, jdx, and dx for NMR, IR and MS spectra, mzML and RAW files for MS spectra and FID or ZIP files for NMR spectra. If ChemSpectra is integrated to a work environment such as an ELN or repository server, the chem-spectra-app is a microservice which is in charge of all spectra-related processes, excluding storage and management (which are the main requests gated by the host web application).

Spectra editor and control panel for one dimensional ^1H NMR and ^{13}C NMR data

As the different analysis types need different actions to edit the corresponding data, ChemSpectra enables analysis specific actions in the react-spectra-editor UI. In the case of NMR data, these specific actions are the addition and/or removal of peaks, integration of signals, coupling constant and multiplicity calculation and assignment. Multiplicities are automatically inferred by known libraries^[1] and are checked by additional rules to ensure the correctness of the results. The generated information such as the identified signals, coupling constants and multiplicity can be summarized in form of a signal list. Additionally, the ChemSpectra editor offers a list of the most common reference solvent shifts for ^1H and ^{13}C NMR spectra, allowing the correction of the values given by default.

Main and control panel for IR data

The IR editor and control panel offer the general three functions given for all types of spectra: adding and removing peaks, including an overview about the added and removed signals, and additionally an option to extract the given signals. Corresponding to the reporting standards for IR spectroscopy, the intensity of the identified signals (vw, w, m, s, vs) can be added to the wavenumber that is recorded. The current implementation gives the information in brackets after the corresponding wavenumber (see Fig. 4 for IR spectra in the stand-alone software).

Main and control panel for mass data

Mass spectra differ from NMR and IR spectra in that way, that they may consist of different scans for one measurement. Depending on the internal procedure of an institution, mass spectra – if they are provided digitally - are either provided as original file including all scans that were measured or are provided as one preselected scan. The ChemSpectra control panel for mass spectrometry offers therefore a dropdown menu including a list of the scans that are provided with each file. Per default, the first scan is visualized by the editor, but the user can change this setting to any scan that is more suitable for the analysis. Figure 3 illustrates the functions of ChemSpectra for mass spectra with the example 4-oxo-4H-chromene-3-carbaldehyde. The threshold line can be used to select the signals and the unselected peaks become grey to be clearly distinguishable from the selected ones. Individual signals can be selected to show the m/z value and intensity of the signal. The example shown in Fig. 3 was gained with the file format RAW (recorded with the ThermoFischer instrument QExactive Plus) demonstrating that the editor can be adapted to read and process also proprietary file formats. With respect to the FAIR data principles, data storage and further processing of proprietary file formats is not a preferred or desired procedure but in some cases, alternatives to the use of proprietary files are currently missing. Therefore, spectra editors should offer options to cover this need if possible. ChemSpectra was used to read data files of two instruments of ThermoFischer (model QExactive Plus (ESI) and Thermo Finnigan Mat 95 (EI and FAB mode)) as test-cases for data that are not given in an open file format. Since the RAW file format contains binary data, it has to be decoded before the processing with ChemSpectra is possible. For this purpose, msConvert in Proteowizard is employed to convert MS files from RAW to mzML. MsConvert in a docker container is called by the chem-spectra-app to achieve this job. mzML files are converted to JCAMP-DX using pymzML, an Open Source python mass spectrometry file parser.

Figure 3. Overview of the react-spectra-editor for mass spectrometry data applied to the file format RAW that was recorded with a ThermoFischer Modell QExactive Plus mass spectrometer for commercial 4-oxo-4H-chromene-3-carbaldehyde. The selection options for the available scans are shown.

Stand-alone version

In its stand-alone version, ChemSpectra runs with an additional UI –the chem-spectra-client– that allows the user to add input or retrieve output as an alternative to a connection of other resources or target systems. The chem-spectra-client UI provides three functions in addition to the react-spectra-editor: (1) a

file management to upload the data to be visualized/analyzed, (2) notifications to the user and (3) a text output that is generated to be copied for a further use of the generated data (Fig. 4).

Figure 4. The ChemSpectra UI as stand-alone version used for the visualization of an IR spectrum of commercial 4-oxo-4H-chromene-3-carbaldehyde, uploaded as JCAMP-DX (*.dx) file. Highlighted are the UI parts (A) file upload and user input panel for the analysis and editing of spectra, (B) react-spectra-editor and additional information to view the results and (C) text output panel. The notifications are not shown.

Embedded implementation

Depending on the desired interactions, the implementation with another web application such as an ELN or repository requires additional efforts for system-specific adaptations. For an implementation with *Chemotion* ELN, which is described here exemplarily, different work processes of the ELN have to be merged with ChemSpectra including the direct use of files that were generated from analytical instruments. In addition, challenges such as data persistence, supporting the storage of data for the full data life cycle, and a workflow management need to be considered. In this regard, embedding ChemSpectra into the *Chemotion* ELN was realized by keeping the original input files and adding the newly composed files as persisted data. The original files stored without any modifications are an important resource for any future referencing issues, while the composed files are kept to avoid the need for repeated analyses. Additionally, two images are generated: a low resolution thumbnail for preview and a higher resolution for the reuse for example in publications. Both images are regenerated every time a user edits a spectrum. The embedding of the ChemSpectra editor results in the availability of a set of file formats that can be generated fully automatically without the input of the user or edited if further actions are desired. Figure 5 shows with the example of an ^1H NMR spectroscopy file, how the implementation with the analysis section of *Chemotion* ELN is realized, giving three relevant files for the user: an original file (*.zip), a user-edited version (*.edit.jdx) and an image file (*.edit.png). A direct benefit of the implementation of ChemSpectra with an ELN is the transfer of the gained data analysis to the ELN. This allows the fast analysis of spectroscopic data and the fast and error-free documentation of the obtained results.

Discussion Of The Limitations Of The Current Developments

ChemSpectra was developed as a basis to reach independence from commercial software for standard analytical measurements. The software allows the integration with other web-developments and offers flexibility to cover further analytical techniques with forthcoming extensions. In the current version, the focus of the developments lay on the definition of a basic functionality that covers the most important needs for NMR, IR and MS spectra analysis and the design of a model for smart integration to web based information systems. The software does not offer a comprehensive solution to special types of measurement yet and lacks certain functions compared to established and specialized software. Considering for example NMR analyses in Organic Chemistry, further improvements of the editor should contain functions for the viewing and editing of 2D data and functions for the comparison of different

spectra in one application window. Additionally, the processing of FID files has to be improved by adding advanced phase correction and baseline correction methods (see Supplemental Information for an example). For mass spectrometry analysis, the chem-spectra-app should be extended to support further MS file formats. Initiatives like OpenChrom show how extensive but also successful such a project is when the given challenges are to be solved by the community. The compatibility of ChemSpectra with additional file types and also types of analytical measurements will be an important extension of ChemSpectra in future, building a framework for interoperable analytical data and its use in full compliance to FAIR principles.

Conclusion

ChemSpectra is a software to swiftly visualize and analyze analytical data, integrating solutions for IR (infrared spectroscopy), mass spectrometry (MS), and one-dimensional ^1H and ^{13}C NMR (proton and carbon nuclear magnetic resonance) spectroscopy data. It serves as decentralized work-instrument for the analysis of the most often used types of spectroscopic data in synthetic (organic) chemistry research, being able to deal with the open file formats JCAMP-DX (IR, NMR, MS) and mzML (MS). The software is offered as an Open Source to allow the further extension to other file formats by the community as exemplarily shown for mass spectra files of the type RAW and NMR spectroscopy files of type FID gained from common analytical instruments. All data files that are provided as non-JCAMP-DX files are processed and converted to JCAMP-DX, allowing a standardized treatment of all data files after a first processing step. ChemSpectra is provided in two versions, as standalone version to be used as independent service and as an integrated editor for the *Chemotion* web applications electronic lab notebook (ELN) and repository. The embedded ChemSpectra editor allows the storage of the original spectra along with edited versions, the automatic peak detection according to a default or manually defined threshold and the storage of an automatically generated image of the spectra in png format. To maximize the benefit of the embedded editor for users, a workflow to write the automatically detected or manually chosen signals was implemented. This allows the direct transfer of information to e.g. the ELN or repository. ChemSpectra consists of different modules that are used to build the core software (chem-spectra-app and react-spectra-editor) and the necessary extensions for its use as stand-alone service (chem-spectra-client). As exemplified with the *Chemotion* ELN and repository implementation, it can be adapted to other work environments. ChemSpectra should serve as a basic software to be extended in future with respect to further data type specific analysis functions and its usability for additional file formats. ChemSpectra is released under the AGPL license to encourage its re-use and further developments by the community.

Abbreviations

ELN, Electronic Laboratory Notebook; UI, User Interface; JCAMP-DX, Joint Committee on Atomic and Molecular Physical data extension; NMR, Nuclear Magnetic Resonance; IR, Infrared; MS, Mass Spectrometry; AGPL, Affero General Public License.

Availability And Requirements

Project name: ChemSpectra

Project homepage: eln.chemotion.net

Project demo page: <https://eln.chemotion.net/chemspectra-editor>, additional videos:
<https://github.com/ComPlat/react-spectra-editor>

Operating system(s): platform independent access, developed/tested on Linux and Mac, deployed on Linux.

Library requirements (for developer): (1) User interface: react-spectra-editor (and dependencies: React.js[i]; jcampconverter[ii] (jcamp processing in JavaScript). (2) Peak-picking & jcamp processing: chem-spectra-app (and dependencies: flask (web server) [iii], nmrglue (spectra reading)[iv], pymzML[v] (mzML to JCAMP-DX conversion), matplotlib (image processing), numpy[vi], and scipy (peak-picking)[vii]). (3) Coupling constant and multiplicity inference: cheminfo-js/spectra.³⁰ (4) RAW to mzML conversion: docker and msConvert in Proteowizard²⁹, numpy³², scipy^{38,19} and Matplotlib[viii].

Other requirements: Modern internet browser supporting HTML5 and JavaScript.

Recommended browsers: Chrome

Programming language: JavaScript, Python

Source Code on Github: Chem-spectra-app (<https://github.com/ComPlat/chem-spectra-app>) chem-spectra-client (<https://github.com/ComPlat/chem-spectra-client>), and react-spectra-including demos and videos (<https://github.com/ComPlat/react-spectra-editor>) can be retrieved from github.

License: AGPLv3

[i]<https://github.com/facebook/react/>, last accessed: 11/13/2019

[ii]<https://github.com/cheminfo-js/jcampconverter>, last accessed: 11/14/2019

[iii]<https://github.com/pallets/flask>, last accessed: 11/13/2019

[iv]<https://github.com/jjhelmus/nmrglue>, last accessed: 11/13/2019

[v] Kösters M, Leufken J, Schulze S, Sugimoto K, Klein J, Zahedi RP, Hippler M, Leidel SA, Fufezan C (2018) pymzML v2.0: introducing a highly compressed and seekable gzip format. *Bioinformatics* 34:2513–2514.

[vii]<https://github.com/scipy/scipy>, last accessed: 11/13/2019

Declarations

Ethics approval and consent to participate

Not applicable

Consent for publication

Not applicable

Availability of data and material

The Supporting Information contains further documentation on the transformation of spectra with ChemSpectra, Flow charts for processing spectra of different file formats, decoding and parsing of mass spec, NMR and IR data files. Also, the communication overview with the ELN environment is given for the ELN-embedded ChemSpectra software. Examples for a processing of JCAMP-dx files and fid files in comparison are given.

Competing interests

The authors declare no competing interests.

Funding

This project has been funded by the German Research Foundation (Deutsche Forschungsgemeinschaft, 266379491) and the Ministry of Science, Research and Arts Baden-Württemberg (Ministerium für Wissenschaft, Forschung und Kunst Baden-Württemberg) through the Science Data Center MoMaF. We acknowledge the support by the VirtMat research consortium at the KIT.

Authors' contributions

Yu-Chieh Huang developed the herein described work concerning frontend and backend developments and adapted the ChemSpectra software to the additional challenges that were provided by the corresponding authors. Pierre Tremouilhac and An Nguyen added valuable input and suggestions in discussions. Nicole Jung and Stefan Bräse supervised the development and supported with the interpretation of chemistry specific problems.

Acknowledgments

We acknowledge the support by Deutsche Forschungsgemeinschaft and Open Access Publishing Fund of Karlsruhe Institute of Technology. This work was supported by the Helmholtz program Biointerfaces in Technology and Medicine (BIFTM) and by bwUniCluster, bwFORCluster. For computational resources we acknowledge the bwCloud (<https://www.bw-cloud.org>), funded by the Ministry of Science, Research and Arts Baden-Württemberg (Ministerium für Wissenschaft, Forschung und Kunst Baden-Württemberg).

Authors' information

Not applicable

References

1. <https://pubs.acs.org/doi/abs/10.1021/ja906709t>, last accessed: 11/14/2019
2. <https://docs.chemaxon.com/display/docs/NMR+Predictor#NMRPredictor-NMRSpectrumPreviewPanel>, last accessed: 11/13/2019
3. Link to free TopSpin download: <https://www.bruker.com/service/support-upgrades/software-downloads/nmr/free-topspin-processing/nmr-topspin-license-for-academia.html>
4. <https://www.ffmpeg2.de/spectragryph/>, last accessed: 11/13/2019
5. Lancashire RJ (2007) The JSpecView Project: an Open Source Java viewer and converter for JCAMP-DX, and XML spectral data files. *Chem Cent J* 1:31.
6. <http://wwwchem.uwimona.edu.jm/software/jcampdx.html>, last accessed: 11/13/2019
7. Mohamed A, Nguyen CH, Mamitsuka H (2016) NMRPro: an integrated web component for interactive processing and visualization of NMR spectra. *Bioinformatics* 32:2067–2068.
8. Xia J, Mandal R, Sinelnikov IV, Broadhurst D, Wishart DS (2012) MetaboAnalyst 2.0—a comprehensive server for metabolomic data analysis. *Nucleic Acids Res*, 40:W127–W133.
9. Tulpan D, Léger S, Belliveau L, Culf A, Čuperlović-Culf M (2011) MetaboHunter: an automatic approach for identification of metabolites from ¹H-NMR spectra of complex mixtures. *BMC Bioinf* 12:400.
10. Zhang F, Brüschweiler R (2007) Robust deconvolution of complex mixtures by covariance TOCSY spectroscopy. *Angew Chem Int Ed* 46:2639–2642.
11. Vosegaard T (2015) jsNMR: an embedded platform-independent NMR spectrum viewer. *Magn Reson Chem*, 53:285–290.
12. Beisken S, Conesa P, Haug K, Salek RM, Steinbeck C (2015) SpeckTackle: JavaScript charts for spectroscopy. *J Cheminform* 7:17.
13. <https://www.ebi.ac.uk/metabolights/index>
14. Wishart DS, Jewison T, Guo AC, et al (2012) HMDB 3.0—The Human Metabolome Database in 2013. *Nucleic Acids Res* 41:D801-D807.

15. <https://www.drugbank.ca/>, last accessed: 11/14/2019
16. Wishart DS, Feunang YD, Guo AC, et al (2017) DrugBank 5.0: a major update to the DrugBank database for 2018. *Nucleic Acids Res* 46(D1):D1074-D1082.
17. <https://github.com/cheminfo-js/jcampconverter#readme>, last accessed: 11/13/2019
18. <https://github.com/ComPlat/nmrglue/commits/show-all-data>, last accessed: 11/13/2019
19. Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, Burovski E, Peterson P, Weckesser W, Bright J, van der Walt SJ, Brett M, Wilson J, Millman KJ, Mayorov N, Nelson ARJ, Jones E, Kern R, Larson E, Carey CJ, Polat I, Feng Y, Moore EW, VanderPlas J, Laxalde D, Perktold J, Cimrman R, Henriksen I, Quintero EA, Harris CR, Archibald AM, Ribeiro AH, Pedregosa F, van Mulbregt P, SciPy 1.0 Contributors. (2019) SciPy 1.0–Fundamental Algorithms for Scientific Computing in Python. preprint arXiv:1907.10121
20. Tremouilhac P, Nguyen A, Huang Y-C, Kotov, S, Lütjohann, DS, Hübsch F, Jung N, Bräse S (2017) Chemotion ELN: an Open Source electronic lab notebook for chemists in academia. *J Cheminform* 9:54.
21. Tremouilhac P, Lin C-L, Huang P-C, Huang Y-C, Nguyen A, Jung N, Bach F, Neumair B, Streit A, Bräse S (2020) The Repository Chemotion: Infrastructure for Sustainable Research in Chemistry. *ChemRxiv*. Preprint. <https://doi.org/10.26434/chemrxiv.12195318.v1>
22. Potthoff J, Tremouilhac P, Hodapp P, Neumair B, Bräse S, Jung N (2019) Procedures for systematic capture and management of analytical data in academia. *Analytica Chimica Acta: X*, 1, Article: 100007.
23. <https://github.com/ComPlat/chem-spectra-app>, last accessed: 11/13/2019
24. <https://github.com/ComPlat/chem-spectra-client>, last accessed: 11/13/2019
25. <https://github.com/ComPlat/react-spectra-editor>, last accessed: 11/13/2019
26. Deutsch EW (2010) Mass Spectrometer Output File Format mzML. *Methods Mol Biol* 604:319–331.
27. <https://github.com/cheminfo-js/spectra>
28. Cobas JC, Constantino-Castillo V, Martín-Pastor M, del Río-Portilla F (2005) A two-stage approach to automatic determination of ^1H NMR coupling constants. *Magn Reson Chem.* 43(10):843-848. doi:10.1002/mrc.1623
29. Chambers MC, MacLean B, Burke R, Amode D, Ruderman DL, Neumann S, Gatto L, Fischer B, Pratt B, Egertson J, Hoff K, Kessner D, Tasman N, Shulman N, Frewen B, Baker TA, Brusniak MY, Paulse C, Creasy D, Flashner L, Kani K, Moulding C, Seymour SL, Nuwaysir LM, Lefebvre B, Kuhlmann F, Roark J, Rainer P, Detlev S, Hemenway T, Huhmer A, Langridge J, Connolly B, Chadick T, Holly K, Eckels J, Deutsch EW, Moritz RL, Katz JE, Agus DB, MacCoss M, Tabb DL, Mallick P (2012) A cross-platform toolkit for mass spectrometry and proteomics. *Nature Biotechnology* 30:918-920.
30. A forth file format, *.infer.json, is generated for internal processes and data comparison.
31. <https://lablicate.com/platform/openchrom>, last accessed: 11/14/2019
32. <https://github.com/facebook/react/>, last accessed: 11/13/2019

33. <https://github.com/cheminfo-js/jcampconverter>, last accessed: 11/14/2019
34. <https://github.com/pallets/flask>, last accessed: 11/13/2019
35. <https://github.com/jjhelmus/nmrglue>, last accessed: 11/13/2019
36. Kösters M, Leufken J, Schulze S, Sugimoto K, Klein J, Zahedi RP, Hippler M, Leidel SA, Fufezan C (2018) pymzML v2.0: introducing a highly compressed and seekable gzip format. *Bioinformatics* 34:2513–2514.
37. <https://www.numpy.org/>, last accessed: 11/13/2019
38. <https://github.com/scipy/scipy>, last accessed: 11/13/2019
39. <https://matplotlib.org/>, last accessed: 11/13/2019

Figures

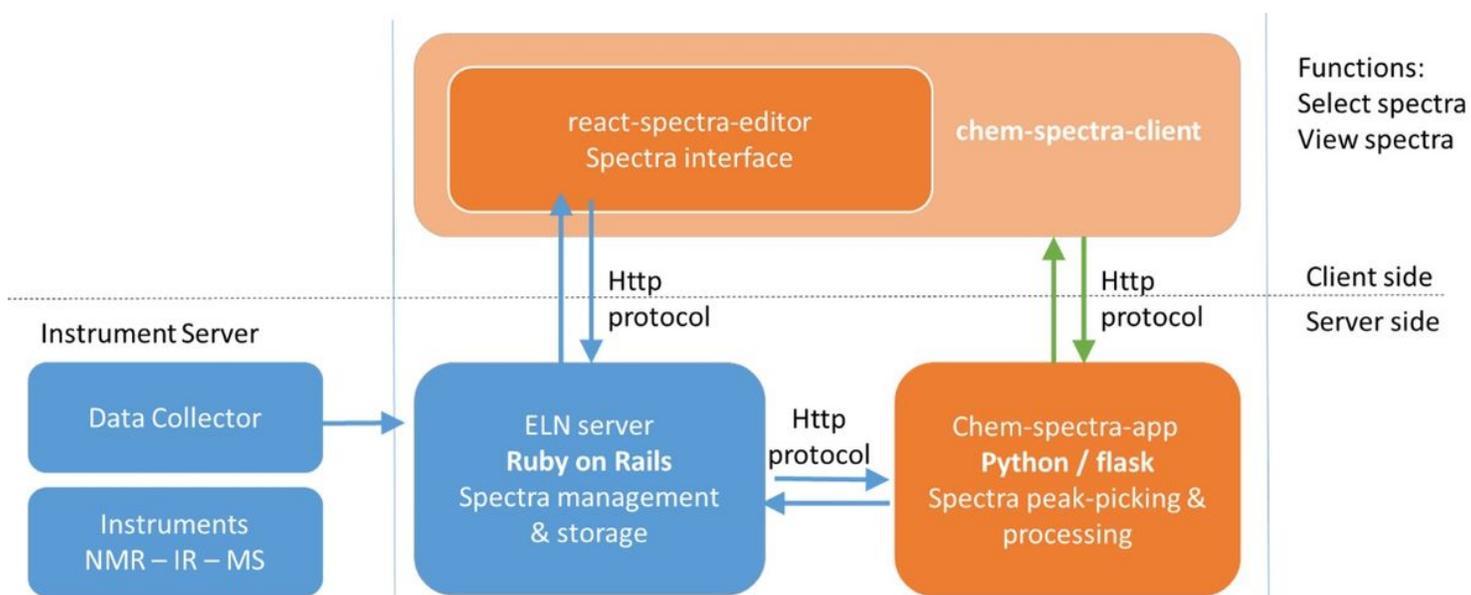


Figure 1

General scheme describing the overall implementation. Orange = necessary parts for a stand-alone version of ChemSpectra, blue: implementation to Chemotion ELN and connection of analysis results via an instrument server.

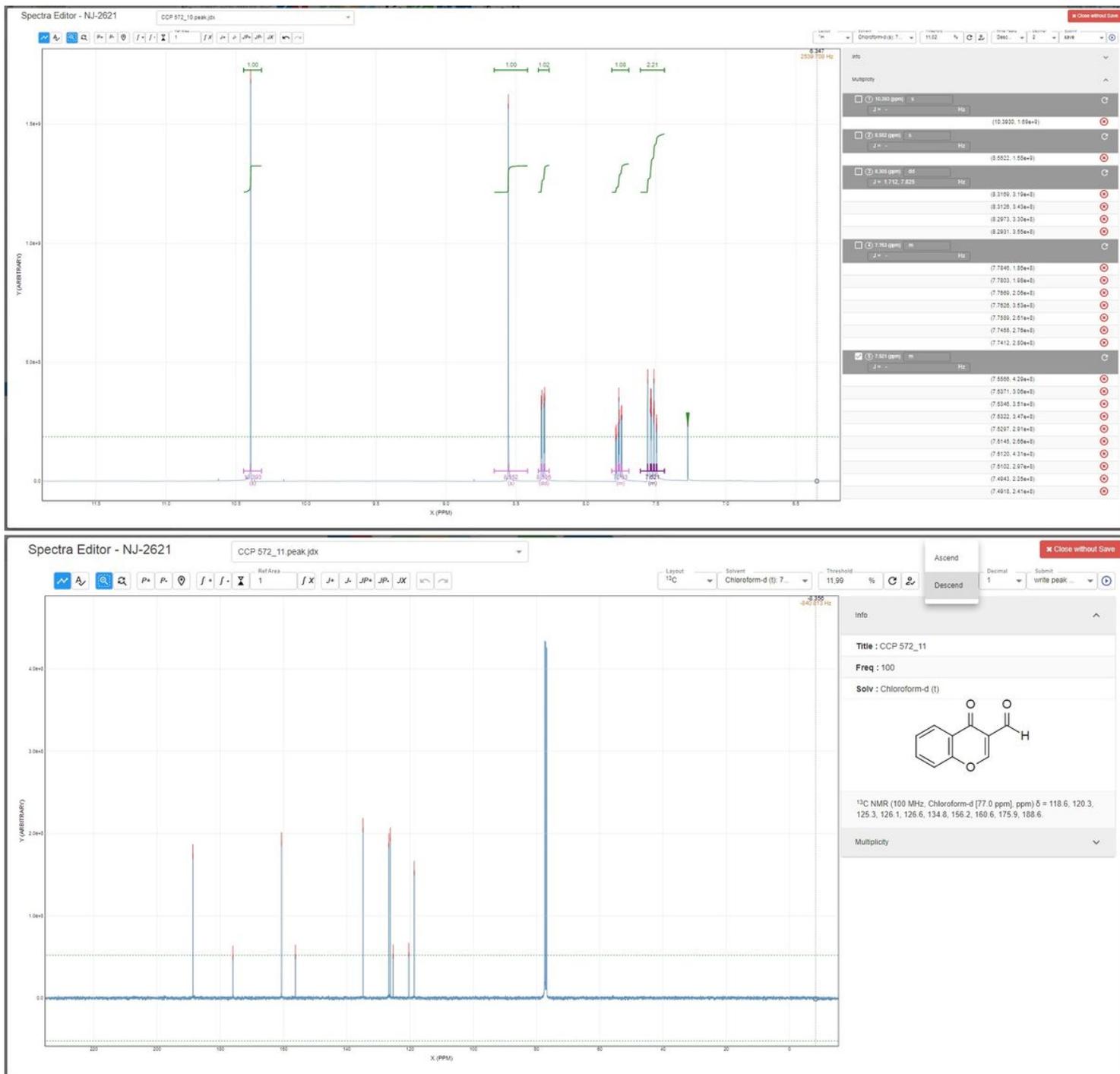


Figure 2

UI of the react-spectra-editor for 1H NMR data (upper part) and 13C NMR data (lower part) generated for a commercial sample of 4-oxo-4H-chromene-3-carbaldehyde measured in CDCl3. Additional information such as the generated peak lists or the information on the structure of the assigned sample is shown in the right information panel. The main toolbar for basic edit functions are given above the spectra visualization panel.

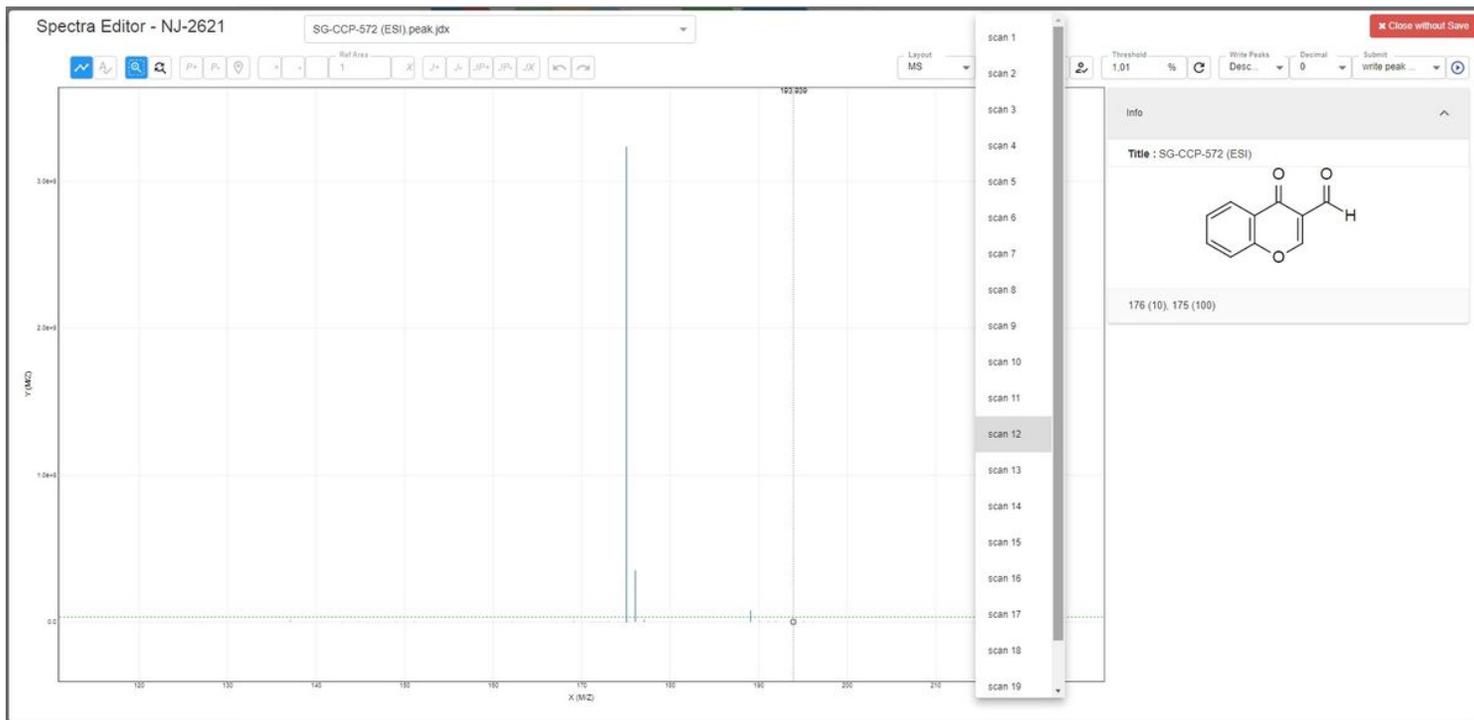


Figure 3

Overview of the react-spectra-editor for mass spectrometry data applied to the file format RAW that was recorded with a ThermoFischer Modell QExactive Plus mass spectrometer for commercial 4-oxo-4H-chromene-3-carbaldehyde. The selection options for the available scans are shown.

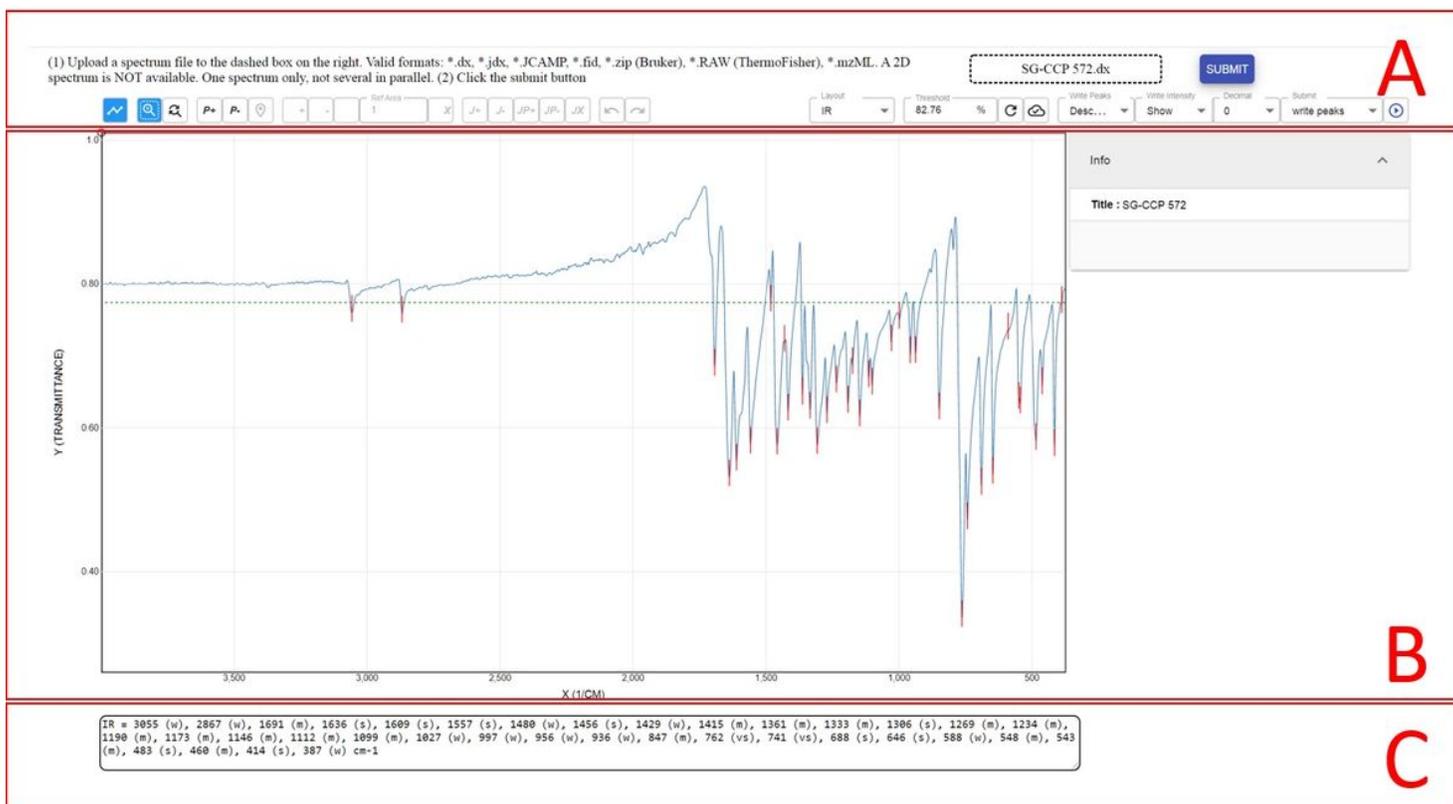


Figure 4

The ChemSpectra UI as stand-alone version used for the visualization of an IR spectrum of commercial 4-oxo-4H-chromene-3-carbaldehyde, uploaded as JCAMP-DX (*.dx) file. Highlighted are the UI parts (A) file upload and user input panel for the analysis and editing of spectra, (B) react-spectra-editor and additional information to view the results and (C) text output panel. The notifications are not shown.

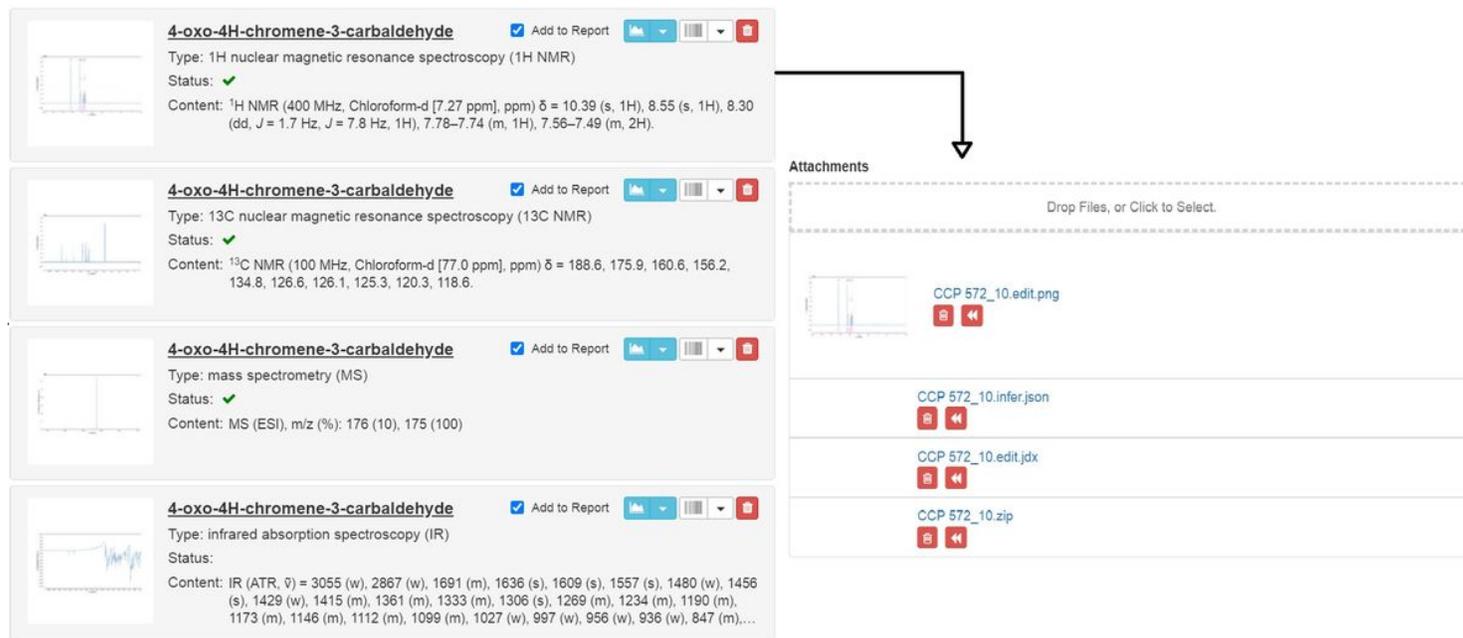


Figure 5

Embedded ChemSpectra in Chemotion ELN: Visualization of data preview in the analysis section of the ELN given for 1H and 13C NMR, MS and IR data (left panel). The peak lists can be transferred directly from react-spectra-editor UI to the ELN. The right panel gives an overview of the automatically generated file types for one uploaded 1H NMR zip file (including a processable fid file) provided by a Bruker 400 MHz instrument.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [Examplefortesting1HNMR.dx](#)
- [SI2020.docx](#)