

integrated community-driven metadata models for neuroscience graph databases

by Lyuba Zehl

Jülich Research Center | Institute of Neuroscience and Medicine - Structural & functional organisation of the brain (INM-1) | Big Data Analytics Co-lead of the EBRAINS Curation Service | PO & Lead developer of openMINDS











openMINDS is the overall umbrella of interlinked metadata models for neuroscience graph databases.







openMINDS is the overall umbrella of interlinked metadata models for neuroscience graph databases.

Each metadata model consists of **modular metadata schemas** that define descriptions of neuroscience research products.







openMINDS is the overall umbrella of interlinked metadata models for neuroscience graph databases.

Each metadata model consists of **modular metadata** schemas that define descriptions of neuroscience research products.

Research products are currently **datasets**, **computer models**, **software**, (meta)data models & brain atlases.







openMINDS is the overall umbrella of interlinked metadata models for neuroscience graph databases.

Each metadata model consists of **modular metadata** schemas that define descriptions of neuroscience research products.

Research products are currently **datasets**, **computer models**, **software**, (meta)data models & brain atlases.



[1] https://github.com/HumanBrainProject/openMINDS_core
 → general origin, resource location, and content of research products





openMINDS is the overall umbrella of interlinked metadata models for neuroscience graph databases.

Each metadata model consists of **modular metadata** schemas that define descriptions of neuroscience research products.

Research products are currently **datasets**, **computer models**, **software**, (meta)data models & brain atlases.



- [1] https://github.com/HumanBrainProject/openMINDS_core
- \rightarrow general origin, resource location, and content of research products
- [2] https://github.com/HumanBrainProject/openMINDS_SANDS
- \rightarrow anatomical location; identification of brain atlases & coordinate spaces







openMINDS is the overall umbrella of interlinked metadata models for neuroscience graph databases.

Each metadata model consists of **modular metadata** schemas that define descriptions of neuroscience research products.

Research products are currently **datasets**, **computer models**, **software**, (meta)data models & brain atlases.



- [1] https://github.com/HumanBrainProject/openMINDS_core
- \rightarrow general origin, resource location, and content of research products
- [2] https://github.com/HumanBrainProject/openMINDS_SANDS
- \rightarrow anatomical location; identification of brain atlases & coordinate spaces
- [3] https://github.com/HumanBrainProject/openMINDS_controlledTerms
- \rightarrow terminology that should be reused across research products





openMINDS is the overall umbrella of interlinked metadata models for neuroscience graph databases.

Each metadata model consists of **modular metadata** schemas that define descriptions of neuroscience research products.

Research products are currently **datasets**, **computer models**, **software**, (meta)data models & brain atlases.



- [1] https://github.com/HumanBrainProject/openMINDS_core
- \rightarrow general origin, resource location, and content of research products
- [2] https://github.com/HumanBrainProject/openMINDS_SANDS
- \rightarrow anatomical location; identification of brain atlases & coordinate spaces
- [3] https://github.com/HumanBrainProject/openMINDS_controlledTerms
- \rightarrow terminology that should be reused across research products



 \rightarrow [in dev.] provenance of simulations, data analysis and visualizations



openMINDS is the overall umbrella of interlinked metadata models for neuroscience graph databases.

Each metadata model consists of **modular metadata** schemas that define descriptions of neuroscience research products.

Research products are currently **datasets**, **computer models**, **software**, (meta)data models & brain atlases.





- [1] https://github.com/HumanBrainProject/openMINDS_core
- \rightarrow general origin, resource location, and content of research products
- [2] https://github.com/HumanBrainProject/openMINDS_SANDS
- \rightarrow anatomical location; identification of brain atlases & coordinate spaces
- [3] https://github.com/HumanBrainProject/openMINDS_controlledTerms
- \rightarrow terminology that should be reused across research products

- [4] https://github.com/HumanBrainProject/openMINDS_computation
- \rightarrow [in dev.] provenance of simulations, data analysis and visualizations
- [5] https://github.com/HumanBrainProject/openMINDS_ephys
- \rightarrow [in dev.] provenance of electrophysiological experiments



openMINDS is the overall umbrella of interlinked metadata models for neuroscience graph databases.

Each metadata model consists of **modular metadata** schemas that define descriptions of neuroscience research products.

Research products are currently **datasets**, **computer models**, **software**, (meta)data models & brain atlases.





- [1] https://github.com/HumanBrainProject/openMINDS_core
- \rightarrow general origin, resource location, and content of research products
- [2] https://github.com/HumanBrainProject/openMINDS_SANDS
- \rightarrow anatomical location; identification of brain atlases & coordinate spaces
- [3] https://github.com/HumanBrainProject/openMINDS_controlledTerms
- \rightarrow terminology that should be reused across research products

- [4] https://github.com/HumanBrainProject/openMINDS_computation
- \rightarrow [in dev.] provenance of simulations, data analysis and visualizations
- [5] https://github.com/HumanBrainProject/openMINDS_ephys
- \rightarrow [in dev.] provenance of electrophysiological experiments
- [6] https://github.com/HumanBrainProject/openMINDS_???
- → potential new extensions (metadata models) to openMINDS



openMINDS is the overall umbrella of interlinked metadata models for neuroscience graph databases.

Each metadata model consists of **modular metadata schemas** that define descriptions of neuroscience research products.

Research products are currently **datasets**, **computer models**, **software**, (meta)data models & brain atlases.



https://github.com/HumanBrainProject/openMINDS
→ central integration & access point for all metadata models
→ central maintenance point for the openMINDS vocabulary
→ provision of all schemas in formal, well-known formats



31

oper

[5]

Pen MND



oper

2]

- [1] https://github.com/HumanBrainProject/openMINDS_core
- \rightarrow general origin, resource location, and content of research products
- [2] https://github.com/HumanBrainProject/openMINDS_SANDS
- \rightarrow anatomical location; identification of brain atlases & coordinate spaces
- [3] https://github.com/HumanBrainProject/openMINDS_controlledTerms
- \rightarrow terminology that should be reused across research products

- [4] https://github.com/HumanBrainProject/openMINDS_computation
- \rightarrow [in dev.] provenance of simulations, data analysis and visualizations
- [5] https://github.com/HumanBrainProject/openMINDS_ephys
- \rightarrow [in dev.] provenance of electrophysiological experiments

oper

[4]

- [6] https://github.com/HumanBrainProject/openMINDS_???
- \rightarrow potential new extensions (metadata models) to openMINDS



61

openMINDS - technical implementation





openMINDS syntax

{" type": "https://openminds.ebrains.eu/core/Person", "required": ["givenName"] "properties": { "familyName": { "type": "string". "_instruction": "Enter the family name of this person."}, "givenName": { "type": "string" "_instruction": "Enter the given name of this person."}}}

 \rightarrow common technical attributes & vocabulary definitions are outsourced

JSON-Schema

{"\$schema": "http://json-schema.org/draft-07/schema#", "\$id": "https://openminds.ebrains.eu/core/Person", "type": "object", "title": "Person". "description": "Structured information on a person.", "required": ["@id", "@type", "https://openminds.ebrains.eu/vocab/givenName"] "properties": { "@id": { "type": "string", "description": "Metadata node identifier."}, "@type": { "type": "string", "const": "https://openminds.ebrains.eu/core/Person"}, "https://openminds.ebrains.eu/vocab/familyName": { "type": "string", "title": "familvName". "description": "Name borne in common by members of a family.", " instruction": "Enter the family name of this person." }, "https://openminds.ebrains.eu/vocab/givenName": { "type": "string", "title": "givenName", "description": "Name given to a person (without family name).", "_instruction": "Enter the given name of this person."}}}

[Example: Definition of the Person schema (snippet) written in openMINDS syntax and its translation to JSON-Schema through the openMINDS integration pipeline.]





openMINDS syntax

{"_type": "https://openminds.ebrains.eu/core/Person",
 "required": [
 "givenName"]
 "properties": {
 "familyName": {
 "type": "string",
 "_instruction": "Enter the family name of this person."},
 "givenName": {
 "type": "string",
 "_instruction": "Enter the given name of this person."}}}

→ common technical attributes & vocabulary definitions are outsourced



JSON-Schema

{"\$schema": "http://json-schema.org/draft-07/schema#", "\$id": "https://openminds.ebrains.eu/core/Person", "type": "object", "title": "Person". "description": "Structured information on a person.", "required": ["@id", "@type", "https://openminds.ebrains.eu/vocab/givenName"] "properties": { "@id": { "type": "string", "description": "Metadata node identifier."}, "@tvpe": { "type": "string", "const": "https://openminds.ebrains.eu/core/Person"}, "https://openminds.ebrains.eu/vocab/familyName": { "type": "string", "title": "familvName". "description": "Name borne in common by members of a family.", " instruction": "Enter the family name of this person." }, "https://openminds.ebrains.eu/vocab/givenName": { "type": "string", "title": "givenName", "description": "Name given to a person (without family name).", "_instruction": "Enter the given name of this person."}}}

[Example: Definition of the Person schema (snippet) written in openMINDS syntax and its translation to JSON-Schema through the openMINDS integration pipeline.]





openMINDS syntax

- {"_type": "https://openminds.ebrains.eu/core/Person",
 "required": [
 "givenName"]
 "properties": {
 "familyName": {
 "type": "string",
 "_instruction": "Enter the family name of this person."},
 "givenName": {
 "type": "string"
 "_instruction": "Enter the given name of this person."}}}
- → common technical attributes & vocabulary definitions are outsourced
- → introduction of additional attributes unknown to formal schema formats



JSON-Schema

{"\$schema": "http://json-schema.org/draft-07/schema#", "\$id": "https://openminds.ebrains.eu/core/Person", "type": "object", "title": "Person". "description": "Structured information on a person.", "required": ["@id", "@type", "https://openminds.ebrains.eu/vocab/givenName"] "properties": { "@id": { "type": "string", "description": "Metadata node identifier."}, "@tvpe": { "type": "string", "const": "https://openminds.ebrains.eu/core/Person"}, "https://openminds.ebrains.eu/vocab/familyName": { "type": "string", "title": "familvName". "description": "Name borne in common by members of a family.", " instruction": "Enter the family name of this person." }, "https://openminds.ebrains.eu/vocab/givenName": { "type": "string", "title": "givenName", "description": "Name given to a person (without family name).", "_instruction": "Enter the given name of this person."}}}

[Example: Definition of the Person schema (snippet) written in openMINDS syntax and its translation to JSON-Schema through the openMINDS integration pipeline.]





openMINDS syntax

- {"_type": "https://openminds.ebrains.eu/core/Person",
 "required": [
 "givenName"]
 "properties": {
 "familyName": {
 "type": "string",
 "_instruction": "Enter the family name of this person."},
 "givenName": {
 "type": "string"
 "_instruction": "Enter the given name of this person."}}}
- common technical attributes & vocabulary definitions are outsourced
 introduction of additional attributes unknown to formal schema formats
 allows to define & maintain vocabulary centrally
 /vocab

JSON-Schema

{"\$schema": "http://json-schema.org/draft-07/schema#", "\$id": "https://openminds.ebrains.eu/core/Person", "type": "object", "title": "Person". "description": "Structured information on a person.", "required": ["@id", "@type", "https://openminds.ebrains.eu/vocab/givenName"] "properties": { "@id": { "type": "string", "description": "Metadata node identifier."}, "@tvpe": { "type": "string", "const": "https://openminds.ebrains.eu/core/Person"}, "https://openminds.ebrains.eu/vocab/familyName": { "type": "string", "title": "familvName". "description": "Name borne in common by members of a family.", " instruction": "Enter the family name of this person." }, "https://openminds.ebrains.eu/vocab/givenName": { "type": "string", "title": "givenName", "description": "Name given to a person (without family name).", " instruction": "Enter the given name of this person."}}}

[Example: Definition of the Person schema (snippet) written in openMINDS syntax and its translation to JSON-Schema through the openMINDS integration pipeline.]







🛱 HumanB	rainProject / op	enMINDS 💿	Watch 👻 11	r Star 11 vy Fork	5
<> Code	⊙ Issues 8 ;	ງ Pull requests 🕟 Actions	Projects	🗇 Wiki 🛛 🔐	•
ᢞ main ▾	openMINDS / voo	cab /	Go to	file Add file -	•
openMIN	NDS and openMINDS	Automated vocabulary extraction		15 hours ago 🕚 History	y
🗋 propertie	es.json	Automated vocabulary extracti	on	15 hours ago	C
🗋 types.jso	n	switch to "label" instead of "na	me" for types.json	7 days ago	с

→ centrally defined & maintained across schemas, metadata models & versions

 \rightarrow builds automatically, but can be manually enriched over time

openMINDS: ./main/vocab/types.json

```
{...,
```

```
"https://openminds.ebrains.eu/core/Person": {
    "description": "Structured information on a person.",
    "label": "Person",
    "schemas": [
        "core/v0/actors/person",
        "core/v3/actors/person"],
    "translatableTo": [
        "https://schema.org/Person"]},
...}
```

```
{...,
    "https://openminds.ebrains.eu/vocab/contributor": {
        "description": "Legal person that gave or supplied something.",
        "label": "Contributor",
        "labelForReverseLink": "Is contributor of",
        "linkedTypes": [
            "https://openminds.ebrains.eu/core/Organization",
            "https://openminds.ebrains.eu/core/Person"],
        "name": "contributor",
        "sameAs": [
            "https://schema.org/contributor"],
        "schemas": [
            "core/v0/actors/contribution"],
        ...}
```



📮 HumanBra	ainProject / ope	enMINDS	⊙ Watch ▼ 1	1 🏠 Star	11 & Fork 5
<> Code 🧿) Issues 8 11	Pull requests (>) Ad	ctions III Proje	ects 🛄 Wil	<i th="" •••<=""></i>
१° main ▾	openMINDS / voc	ab /		Go to file	Add file - ····
openMIND)S and openMINDS	Automated vocabulary extr	action	15 ho	ours ago 🕚 History
🗋 properties.	json	Automated vocabulary	extraction		15 hours ago
L types.json switch to "label" instead		l of "name" for types	s.json	7 days ago	

→ centrally defined & maintained across schemas, metadata models & versions

- \rightarrow builds automatically, but can be manually enriched over time
- \rightarrow includes general descriptions & human-readable labels

openMINDS: ./main/vocab/types.json

```
{...,
```

"https://openminds.ebrains.eu/core/Person": {
 "description": "Structured information on a person.",
 "label": "Person",
 "schemas": [
 "core/v0/actors/person",
 "core/v3/actors/person"],
 "translatableTo": [
 "https://schema.org/Person"]},
...}

```
{...,
    "https://openminds.ebrains.eu/vocab/contributor": {
        "description": "Legal person that gave or supplied something.",
        "label": "Contributor",
        "labelForReverseLink": "Is contributor of",
        "linkedTypes": [
            "https://openminds.ebrains.eu/core/Organization",
            "https://openminds.ebrains.eu/core/Person"],
        "name": "contributor",
        "sameAs": [
            "https://schema.org/contributor"],
        "schemas": [
            "core/v0/actors/contribution"],
        ...}
```



🛱 HumanBrainProject /	openMINDS 💿 W	atch • 11 🖧 Star 11 😵 Fork	5
<> Code Subscriptson Subscriptso	ୁକ୍ତ Pull requests 🕞 Actions	III Projects 🛄 Wiki	•
ৃ° main ▾ openMINDS /	vocab /	Go to file Add file -	·
openMINDS and openMII	NDS Automated vocabulary extraction		,
🗅 properties.json	Automated vocabulary extraction	15 hours ago	
🗅 types.json	switch to "label" instead of "name	e" for types.json 7 days ago	F

- → centrally defined & maintained across schemas, metadata models & versions
- \rightarrow builds automatically, but can be manually enriched over time
- \rightarrow includes general descriptions & human-readable labels
- \rightarrow includes where/when the type or property is/was used

openMINDS: ./main/vocab/types.json

{...,

```
"https://openminds.ebrains.eu/core/Person": {
    "description": "Structured information on a person.",
    "label": "Person",
    "schemas": [
        "core/v0/actors/person",
        "core/v3/actors/person"],
    "translatableTo": [
        "https://schema.org/Person"]},
...}
```

```
{...,
    "https://openminds.ebrains.eu/vocab/contributor": {
        "description": "Legal person that gave or supplied something.",
        "label": "Contributor",
        "labelForReverseLink": "Is contributor of",
        "linkedTypes": [
            "https://openminds.ebrains.eu/core/Organization",
            "https://openminds.ebrains.eu/core/Person"],
        "name": "contributor",
        "sameAs": [
            "https://schema.org/contributor"],
        "schemas": [
            "core/v0/actors/contribution"]},
...]
```



📮 Humanl	BrainProject / o	penMINDS	⊙ Watch ▼	11 🏠 Star	11 % Fork 5
<> Code	 Issues 8 	ំង Pull requests 🕞	Actions III Proje	ects 🔲 Wiki	
	openMINDS / ve	ocab /		Go to file	Add file - ····
openM	IINDS and openMIND	S Automated vocabulary e	xtraction	15 ho	urs ago 🕚 History
🗋 propert	ties.json	Automated vocabula	ry extraction		15 hours ago
🗋 types.js	son	switch to "label" inste	ad of "name" for type	s.json	7 days ago

- → centrally defined & maintained across schemas, metadata models & versions
- \rightarrow builds automatically, but can be manually enriched over time
- \rightarrow includes general descriptions & human-readable labels
- \rightarrow includes where/when the type or property is/was used
- → enables references to matching schema types or properties of other metadata initiatives

openMINDS: ./main/vocab/types.json

```
{...,
```

```
"https://openminds.ebrains.eu/core/Person": {
    "description": "Structured information on a person.",
    "label": "Person",
    "schemas": [
        "core/v0/actors/person",
        "core/v3/actors/person"],
    "translatableTo": [
        "https://schema.org/Person"]},
...}
```

```
{...,
    "https://openminds.ebrains.eu/vocab/contributor": {
        "description": "Legal person that gave or supplied something.",
        "label": "Contributor",
        "labelForReverseLink": "Is contributor of",
        "linkedTypes": [
            "https://openminds.ebrains.eu/core/Organization",
            "https://openminds.ebrains.eu/core/Person"],
        "name": "contributor",
        "sameAs": [
            "https://schema.org/contributor"],
        "schemas": [
            "core/v0/actors/contribution"]},
...}
```







openMINDS schema model for core, controlledTerms and SANDS

by Ulrike Schlegel









openMINDS schema model for core, controlledTerms and SANDS

openMINDS linked metadata instances (JSONLDs)

{"@context": {
 "@vocab": PROPERTY-NAMESPACE},
 "@type": SCHEMA-TYPE,
 "@id": UNIQUE-IDENTIFIER,
 "PROPERTY-NAME": VALUE}

by Ulrike Schlegel







openMINDS schema model for core, controlledTerms and SANDS

openMINDS linked metadata instances (JSONLDs)

{"@context": {
 "@vocab": "https://openminds.ebrains.eu/vocab/"},
 "@type": SCHEMA-TYPE,
 "@id": UNIQUE-IDENTIFIER,
 PROPERTY-NAME: VALUE}

{"@context": {
 "@vocab": "https://openminds.ebrains.eu/vocab/"},
 "@type": "https://openminds.ebrains.eu/core/ContactInformation",
 "@id": "d72ba653-ac9a-428d-ba65-84c137ce5ad6",
 "email": "l.zehl@fz-juelich.de"}

by Ulrike Schlegel







openMINDS schema model for core, controlledTerms and SANDS



openMINDS linked metadata instances (JSONLDs)

{"@context": {
 "@vocab": "https://openminds.ebrains.eu/vocab/"},
 "@type": SCHEMA-TYPE,
 "@id": UNIQUE-IDENTIFIER,
 PROPERTY-NAME: VALUE}

{"@context": {

"@vocab": "https://openminds.ebrains.eu/vocab/"},
"@type": "https://openminds.ebrains.eu/core/ContactInformation",
"@id": "d72ba653-ac9a-428d-ba65-84c137ce5ad6",
"email": "l.zehl@fz-juelich.de"}

{"@context": {

```
"@vocab": "https://openminds.ebrains.eu/vocab/"},
"@type": "https://openminds.ebrains.eu/core/Person",
"@id": "5bc17d5c-f5a2-4350-be30-8515da2e4e37",
"givenName": "Lyuba",
"familyName": "Zehl",
"contactInformation": {
    "@id": "d72ba653-ac9a-428d-ba65-84c137ce5ad6"}}
```



openMINDS - tooling & support





openMINDS services: Instance libraries for selected schemas



by openMINDS & EBRAINS KG Team

openMINDS Python library & Wizard

https://github.com/HumanBrainProject/openMINDS_generator alpha version
https://pypi.org/project/openMINDS

pip install openminds

import openMINDS
import openMINDS.version_manager

Initialise the local copy of openMINDS
openMINDS.version_manager.init()

Select which version of openMINDS to use
openMINDS.version_manager.version_selection('v2.0.0')

initiate the helper class for the dynamic usage of a specific openMINDS version helper = openMINDS.Helper()

initiate the collection into which you will store all metadata instances
mycollection = helper.create_collection()

create a metadata instance for (e.g.) the openMINDS Person schema
person_open = mycollection.add_core_person(givenName="open")

add more metadata to a created instance
mycollection.get(person_open).familyName = "MINDS"

add connections to other metadata instances email_openminds = mycollection.add_core_contactInformation(email="openminds@ebrains.eu") mycollection.get(person_open).contactInformation = email_openminds

save your collection
mycollection.save("./myFirstOpenMINDSMetadataCollection/")

https://github.com/HumanBrainProject/openMINDS_wizard ^{alpha} version https://wizard.kg-ppd.ebrains.eu ^{(temporary URL;} this might change)



Name		
Description		
Version number		
Accessibility		
	v	
Type		
, i pe	•	
1		
License		
Repository URL		
Keywords		
	+	
Released date		
mm / dd / yyyy		
	Next	

"openMINDS" by Lyuba Zehl, Dr. rer nat. | PO & Lead developer of openMINDS | Nov 2021

Copyright © 2

openMINDS HTML views deployed as GitHub pages

https://humanbrainproject.github.io/openMINDS/



Person

https://openminds.ebrains.eu/core/Person

Download specification

part of core in version v4 - view as JSON Schema

Required properties are highlighted in **bold**

computation

controlledTerms

core

SANDS

PROPERTY		VALUE		
NAME*	DESCRIPTION	ТҮРЕ	INSTRUCTION	
givenName	Name given to a person, including all potential middle names, but excluding the family name.	string	Enter the given name of this person.	
affiliation	Declaration of a person being closely associated to an organization.	embedded object array (1 - n) of type: <u>Affiliation</u>	Add the current and, if necessary, past affiliations of this person.	
contactInformation	Any available way used to contact a person or business (e.g., address, phone number, email address, etc.).	linked object ContactInformation	Add the contact information of this person.	
digitalldentifier	Digital handle to identify objects or legal persons.	linked object array (1 - n) of type: <u>ORCID</u>	Add one or several globally unique and persistent digital identifier for this person.	
familyName	Name borne in common by members of a family.	string	Enter the family name of this person.	

*This is the simplified property name - within a metadata instance of this schema (JSON-LD) the properties are extended to map to the openMINDS vocabulary namespace ("https://openminds.ebrains.eu/vocab/PROPERTY_NAME").





openMINDS on the EBRAINS Collaboratory

Collaboratory	Collabs Documentation	Support Forum Q 🌲 🧖 (
₽ Public	₩ + / Collabs + / openMINDS +	
openMINDS	openMINDS	♥ 2 ♦ Settings 🖋 Edit + Cr
august at	ast modified by Lyuba Zebi on 2021/07/12 1	
Documentation		
Vive		
AOs		
How to contribute		
ab		
Team		
Tutorials		
	The open Metadata Initiative for Neuroscience Dat	ta Structures:
	Community-driven metadata models for neuroscie	nce graph databases.
	 I his collab appeals to members of the neuroscience community that are interested in accordance with openMINDS. 	Jump to
	(2) conceptually contributing to openMINDS	> Documentation
	(3) programmatically contributing to openMINDS	 Metadata models & schemas Application details
	What you can find here:	 Technical details
	+ general overview of openMINDS	 How to contribute Code of conduct
	+ documentation & HTML views for all openMINDS metadata models	 Report & ask questions
	+ documentation for contributing to openMINDS	Direct contributions Tutorials
	the second station of the second station of the second state state of the second state state of the second state stat	
	+ documentation for using openMINDS (Incl. tutonals)	* Presentations
	+ documentation for using openMINDS (incl. tutoriais) Who is behind openMINDS?	Presentations Examples FAQs
	documentation for using openMINUS (incl. tutorials) Who is behind openMINDS? openMINDS is powered by HBP/r (Human Brain Project) and FBPAINS?	Freesentations FAQs
	occumentation for using operativitys (incl. suborals) Who is behind openMINDS? openMINDS is powered by HBP# (Human Brain Project) and EBRAINS#, I external conflutions throughout the neuroscience community.	Freesenations Freesenations Freesenations Freesenations Freesenations Freesenations Freesenations
	toocumentation for using operativities (incl. tutoreals) Who is behind openMINDS? openMINDS is powered by HBPC (Human Brain Project) and EBRAINSP, i seteral contributions throughout the neuroscience community. We the neuroscience community.	Presentations * Examples * FAGs but looking for Supported research produc the ERBAINS Dateset
	* occumentation for Using optimizinus (incl. unionals) Who is behind openMINDS? openMINDS is powered by HBPC Human Brain Project] and EBRAINSER, i external contributions throughout the neuroscience community. We, the openMINDS development team, currently units knowledge from curation Service, the EBRAINS Kines Service of the EBRAINS Kines	Presentations Fixage Fixed Fi
	* occumentation for using optimizing so that the second solution of the second solution of the second solution of the second solution of the second solution second solut	* Preventations * Examples * Examples * FAGs
	* occumentation for using optimizing optimizing and the contrast of the optimizing optimizing optimizing and the optimizing optize optimizing optimize optize optimizing optimize optimize optimi	* Presentations * Compose * Compose * Compose * FAgs
	* occumentation for Using optimizing to the contrast Who is behind openMINDS? openMINDs is powered by HBPE (Human Brain Project) and EBRANSK, it external contributions throughout the neuroscience community. We, the openMINDS development team, currently unite knowledge fram, curation Service, the EBRANS Muscle and the EBRANS Muscle and the Contrast on Service, the EBRANS Muscle and the service development team, currently unite knowledge fram, there are a solution of the service development team, service developmenteam, service development team, service development team, service d	Presentations Coamples FAgs but looking for the EBRAINS the EBRAINS MesBaltabdes ¹ Model ¹ Software ⁴
	* occumentation for using optimizintus (incl. unormati) Who is behind openMINDS: pointMINDS is powered by HBPEr Human Brain Project) and EBRAINSER, t external contributions throughout the neuroscience community. We, the openMINDS development team, currently unite knowledge from curation service, the EBRAINS Movinged orfan, the EBRAINS Mass BrAINE Knowledge Space teams. Here is our support email: openmindsgled active yet). Who adopted openMINDS? Within EBRAINS, the openMINDS metiatata models are adopted by	the EBRANS
	* occumentation for using optimizing on the contrast of the c	Precentations Compose Compose
	* occumentation for Using optimizing to the contrast in the c	
	* occumentation for Using optimizing to the contrast of the c	
	* eccementation for Using optimizinto's line: Lutionality Who is behind oppenMINDS genMINDS is powered by HBP* (Human Brain Project) and EBRAINSER, I external contributions throughout the neuroscience community. We, the openMINDS development team, currently unite knowledge fram curation service, the EBRAINS through the EBRAINS that is be the? Knowledge Space team. Here is out support email: openmindsget active yet). Who adopted oppenNINDS metiadata models are adopted by Knowledge Chapt, the EBRAINS, the openNINDS metiadata models are adopted by Knowledge Chapt, the EBRAINS, the openNINDS metiadata models are adopted by Knowledge Chapt, the EBRAINS, the openNINDS metiadata models are adopted by Knowledge Chapt, the EBRAINS that Service, and the Virtual Brain (TV) openMINDS is currently in the process of being adopted by the Japar projects. For Integrating and maintaining community-whee scoped neuroscience on the clopel's collapation team the line team of the scoped neuroscience on	
	* eocumentation for Using optimizintus (Incl. Unional) ** Ocumentation for Using optimizintus (Incl. Unional) ** Ocumentation (Incl. Incl. Incl	

What you can find here:

→ Documentation of ...

- metadata models & schemas (linked to HTML views)
- application details
- technical details

\rightarrow How to contribute ...

- code of conduct
- report & ask questions
- direct contributions

→ Tutorials ...

- presentations
- examples in prep

→ FAQs

We are always happy to extend / improve the Collab

based on your feedback! 😊



openMINDS Community Forum @ INCF NeuroStars

https://neurostars.org/t/openminds-community-forum-virtual

mincf NeuroStars

openMINDS Community Forum (virtual)

openminds



Mathew_Abrams

34m

The open Metadata Initiative for Neuroscience Data Structures, short openMINDS, is the overall umbrella for a set of integrated metadata models for describing neuroscience research products in graph databases. openMINDS is powered by HBP and EBRAINS, but looking for external contributions throughout the neuroscience community. Within EBRAINS, the openMINDS metadata models are adopted by the EBRAINS Knowledge Graph and Atlas Service. In addition, openMINDS will be supported by The Virtual Brain (TVB) and is currently in the process of being adopted by the Japan Brain/MINDS project. For integrating and maintaining community-wide accepted neuroscience ontologies, we are closely collaborating with the INCF Knowledge Space and the InterLex project (a core component of the SciCrunch infrastructure of the FAIR Data Informatics Lab, formerly known as the Neuroscience Information Framework).

The purpose of this forum is to provide a place for the community to discuss and contribute to openMINDS.

🔗 📕 🥱 Reply

openminds@ebrains.eu

We are looking forward to your questions, discussions & contributions

Share Bookmark





Reply

Who is behind openMINDS?

openMINDS is powered by HBP & EBRAINS, but we are looking for contributions from the whole neuroscience community!

If you want to learn more about openMINDS just get in touch, have a swith us (in-person or via VC),

your openMINDS development team 😊

Who?	Role?	Team affiliation?	Institute affiliation?
Lyuba Zehl	openMINDS product owner	EBRAINS Curation & Atlas Service	Jülich Research Center (INM-1)
Oliver Schmid	KG product owner & openMINDS developer (tools)	EBRAINS Knowledge Graph	EBRAINS AISBL
Andrew Davison	openMINDS developer (models, in-depth, tools)	EBRAINS Curation Service	Université Paris-Saclay, CNRS (NeuroPSI)
Benjamin Weyers	openMINDS developer (software)	EBRAINS Curation Service	University of Trier (Human-Computer Interaction)
Ulrike Schlegel	openMINDS developer (datasets)	EBRAINS Curation Service	University of Oslo (Department of Anatomy, NeSys)
Stefan Köhnen	openMINDS developer (tools, SANDS)	EBRAINS Curation & Atlas Service	Jülich Research Center (INM-1)
Heidi Kleven	openMINDS developer (SANDS)	EBRAINS Curation Service	University of Oslo (Department of Anatomy, NeSys)
Peyman Najafi	openMINDS developer (models, in-depth)	EBRAINS Curation Service	Université Paris-Saclay, CNRS (NeuroPSI)
Jan Gründling	openMINDS developer (software)	EBRAINS Curation Service	University of Trier (Human-Computer Interaction)
Tom Gillespie	ontology developer & openMINDS developer (SANDS)	InterLex Project & Knowledge Space	INCF, University of California, Dep. of Neuroscience
Mathew Abrams	Knowledge Space developer	Knowledge Space	INCF, Karolinska Institutet
Visakh Muraleedharan	Knowledge Space developer	Knowledge Space	INCF, Karolinska Institutet





Some final information ...

Who has adopted openMINDS?

Within EBRAINS, the openMINDS metadata models are adopted by the **Knowledge Graph**, the **Curation** and **Atlas Service**. openMINDS is also supported by **The Virtual Brain (TVB)** and in the process of being adopted by the **Japan Brain/MINDS project**. For integrating and maintaining community-wide accepted neuroscience ontologies, we are closely collaborating with the **InterLex**, a project of the SciCrunch infrastructure of the **FAIR Data Informatics Lab (FDI Lab)**, f.k.a. the Neuroscience Information Framework (NIF).

Acknowledgements:

The metadata model specification and corresponding open source code was developed in part or in whole in the Human Brain Project, funded from the European Union's Horizon 2020 Framework Programme for Research and Innovation under Specific Grant Agreements No. 720270, No. 785907, and No. 945539 (Human Brain Project SGA1, SGA2, and SGA3).

We would also like to thank the whole **EBRAINS Curation Service team**, the whole **EBRAINS Atlas Service team**, and all **EBRAINS data providers** that gave and are still giving us feedback to guide the development of openMINDS.



