

BCI – BRAIN COMPUTER INTERFACE FRAMEWORK (proposal v.4.4h)

The structure of basic and advanced BCI is presented using HSL¹ notation. The advanced approach includes a mapping of brain „engines” (thinking, emotions, info-retrieval, memorizing, communication) into so called „views” (biological, chemical, energetic, geometric, physical, medical, logical, semantic, psychological, mathematical, ontological).

Notation

<!...>	comment	ppppXxxx	item Xxxx with prefix ‘pppp ‘
< >	container	UUUU.xxxx	xxxx belongs to UUUU
≡>	link to outside	& /	conjunctions ‘and’ ‘or’
<def > </def>	start-end of definition	<-> or ↔	bidirectional passive unitary relation 1:1
[..] or {...}	list of assigned items	-> or →	forward unitary passive relation
(..)	list of items	<=>	many to many active relations
xxxx(..)	name of list	<=	backward active relations
:	equivalent name	=>	forward active relations
=	value assignment	..	more
::	belongs to	...	much more
#	number of	?	to be verified

Person definition is presented below to enable better understanding of HML

```

<def PERSON>
  object.nfo[id,sex,birth-data]
  homeaddress[country,site,street,house,flat]
  body[(brain,liver,kidney,joints,..) weight,height,eyes-colour,defects]
  family[gentree,parent,child,son,daughter,
    grandSon,grandDaughter,granMa,granPa]
  emotion[love,hate,satisfaction,frustration,agression,enjoyment,anger]
  psychComplex[fear-of-insupport,regression,inferiority,persecution]
  habit,hobby,profession,
  health[measures,physical-examinations,illness-history],
  role[advisor,consultant,manager,patron,partner,customer,
    supervisor,participator,owner,supplier, ...]
  state[active,inactive,dormant,suspended,aborted,idle,lost,dead,
    homeless,retired,married/divorced/single,ignored]
  place[point, area,everywhere,nowhere]
  life-space[psychological,social,educational,professional,financial]
  behaviour<!flow of processes of the object >
  behaviour.rational[selfrealization,need,satisfaction]
  behaviour[marriage,friendship,career,ilness,aging]
  genotype,phenotype
  olh:=[birth,aging-curve,social_events,health_illness-events,
    educ-events,job-events,critical_events,death]<!object-life-history>
</def>
  
```

¹ This paper is based on HSL Language DOI: 10.13140/RG.2.2.36330.62409 and is related also to participation in IEEE EMBS-P2731 (UT-BCI) WG. This is not an official document of this group but my individual working report.

RESOURCES[BCI-vocabulary,catalogues(medical-devices,communication-protocols, standards(ISO,IEEE,HL7,..),publications)]

SUBSYSTEMS[BRAIN,DEVICES,BCI-COMMUNICATION,BCI-OUTPUT]

<def **BRAIN**>

brainId(personId &/biologicalId,type)

<! personId refers to person's definition in HSL language>

mainpart[forebrain(cerebrum(hemisphere,thalamus,hypothalamus),midbrain,hindbrain]

area[lobe,cortical-region/electrode-placement-code-mapping]

lobe[frontal,parietal,limbic,occipital,temporal]

part[cerebral-cortex,basal,dienceph,brainstem,cerebellum,hippocampus,spinal-cord]

cortical-region[primary-visual,entorhinal,inferior-temporal,orbitofrontal,
lateral-prefrontal,inferior-parietal]

cortex-area[visual,sensory,tactile,auditory, ..]

brain-function[sensory(vision,hearing,smell,touch,...),

mental(association,speech,language-comprehension,coordination,...),

motor(eye-movement,voluntary-movement, ..)]

detail-part[neuron,synapse,receptor,unpaired-electron,neurotransmitter,..]

neuron {[nucleous(mitochondria,membrane,cytoplasm,vesicle,perycarion),

myelin-sheath,schwann-cell,axon,dendrite],

form(multipolar,bipolar,unipolar)]}

</def brain>

<dev **DEVICES** .>

type[medDevice /& medComp],id,name,

signal-aquisition-method,comm(embedded,external),software(embedded,external),

technical-info(manufacturer,first-instal-year, wear%,configuration-settings,

reliability(OEE (MTTF,MTBF,MTTR),operating-instruction/protocol ...),

medical-info[diseases,effects(direct-effects,side-effects..),contraindications,

precautions,patient-state(before,after)],

risk-of-use(risk-classes),conformity(directives-standards,technical-requirements),

signal-aquisition-method[sensor(electrode(type),chip,...),

invasive/semiinvasive/noninvasive),

cortical-region/point-of-contact/placement-code],

medDevice.function(recording,diagnosis,monitoring,treatment,alleviation,...),

medDevice-category[stationary,mobile,

linked-stand-alone(touch,cable,bluetooth,wifi,internet)],

medDevice[triggered-by(event,time,demand, ...),link-mode(<=>,<= ,=>,<←,→)],

mobile-medDevice[smartphone,wearables(smartwatch,smartband,headset,...)],

stationary-medDevice-name(EcoG,EEG,fMRI,fNIRS,INR,MEG,PET,MRT,),

embedded-software(pacemaker,noise-artifact-removal, frequency-filtering, ...),

potential(ERP(P300..),SCP,TTD,...),

noise-artifacts(movement-related,eye-blinking,heart-related,...) ,

medComp(super-comp,mini-comp,micro-comp,bio-comp,bio-hybrid-comp...)

</def Devices>

<def BCI-COMMUNICATION>

bci.resources(bio-neuralnetworks(simulator/emulator),medDevices/comp/bio-hybrid-comp,...)

<! medComp could be universal computer with medical functionality>

bci.category[basic,hybrid(<bci-1>,<bci-2>,...),advanced]

bci.hybrid(multi-brainpatterns,multi-sensory-stimuli,multi-modal-signals,multi-intelligent.).

<def basicBci>

structureBci[SourceObject,stream(message), ReceivingObject]

SourceObject[personId[cortical-region[layer]/electrode-placement-code]]

ReceivingObject[medical-device /& computer]

{stream(<source><receiver/device>,[message[signal[impulse]],flow-type]

parameters[standard(type,id, encryptionCode,securityCode,timeStamp)

extended(timeInterval,duration,populationShape,dynamics,dispersion,
measureUnit,accuracy,disturbance)]

trigger(event/ERP,time,demand,)... }

<!stream is one/multi-channel collection of messages directed to one receiving device>

<!message is one-channel sequence of signals issued at a given time >

<!signal is a set of electrical impulses that flows on groups of active neurons >

message[cp,data]]<!cp:comm:communication protocol>

signal(EEG,MEG,fMRI,NIRS,EOG,EMG,sEMG,ECG,EcoG,..)

signal-type(magnetic,electro-magnetic,electrical,chemical,fluid/blood-flow),

signal.Electrical(potential(SCP,ERP(P300,...))

cp(tcp/ip,udp,i2c,spi,can,hart,ethernet,wifi,rf,bluetooth,nfc,satellite,6lowpan,
mqtt,zigbee,coap,z-wave,uart,modbus,device-embedded, BCI-p,...)

flow {type[cerebral-fluid,cerebral-impulse,blood-flow,...]

activated/triggered by <.> with <value> at <time-point>/when <condition>

finished at <> with <.> when <.> } </def basicBci>

<def advancedBci>

[engine-view-mapping,intrusion,ext-cp <!extended communication protocol>]

mapping(<engine => view> /& <view →engine>,intrusion)

<! engines and views should belong to the same person>

intrusion(actions => <brain>)

<!active mode(=> <=> <=) passive mode(→ ↔ ←)>

view[active/passive (biological,chemical,energetic,geometric,physical,medical,
logical,semantic,psychological,mathematical,ontological)]

<!active view is based on brain neural-network>

engine[thinking,emotions,info-retrieval,memorizing,intuition(trust,love,hate),
communication,total,.....]}

<!framework of mapping Engine => View - some examples >

<def>emotionEngine => psychologicalView

emotionEngine[Lob.frontal(electro-placements/neural network-area)]

*psychologicalView[love,hate,satisfaction,frustration,agression,enjoyment,anger,
fear of insupport,regression,inferiority,persecution] </def>*

<def>totalEngine => medView <!total Engine refers to the whole brain>

totalEngine(brainArea/neural network-area,disease-pattern)

medView(disease(neurodegenerative,neurological,..),injury)

neurodegenerative(dementia/alzheimers,parkinsons,huntingtons,..)

neurological(autism-spectrum,tumor,migraine,multiplesclerosis,epilepsy,stroke,..)

</def>

<def>**totalEngine** =>/ → **ontologicalView**

totalEngine(*brainStructure*,*brainContents*)

ontologicalView(*brainId*,*brainAging*(*neurons*(*dead*,*born*)),*diseasesHistory*,
brainVolume(*curve*,...),*brainUsage*)

</def>

<def>**totalEngine** =>/ → **energeticView**

totalEngine(*brainStructure*,*brainContents*)

energeticView(*brainId*,*power-consumption vs aging*,*power-supply-disturbances*)

</def>

</def *advancedBci*>

</end BCI-COMMUNICATION>

<def **BCI-OUTPUT**><!basic usage>

<list of reports>

<report>(<list of info>)

for each report {

triggered by(event,time,demand,online..)]

report(each-stream,summary(personId,deviceId,customized)

[generated by <application>,

delivered(on-line,periodically,on-demand) to <user> by <channel>

channel(orally-communicated(directly,phone),mail-internet,post,internal-store]

}

</def *bci-output*>

The end of proposal

this paper is licenced as Creative Commons noncommercial-3.0 Unported