

# Brain Computer Interfaces for Inclusion

P. J. McCullagh  
Computing & Engineering,  
University of Ulster  
Shore Road, Jordanstown,  
Co. Antrim BT37 0QB, UK  
Tel: +44 (0)2890368873  
pj.mccullagh@ulster.ac.uk

M.P. Ware  
Computing & Engineering,  
University of Ulster  
Shore Road, Jordanstown,  
Co. Antrim BT37 0QB, UK  
Tel: +44 (0)28 90366045  
mp.ware@ulster.ac.uk

G. Lightbody  
Computing & Engineering,  
University of Ulster  
Shore Road, Jordanstown,  
Co. Antrim BT37 0QB, UK  
g.lightbody@ulster.ac.uk<sup>i</sup>

## ABSTRACT

In this paper, we describe an intelligent graphical user interface (IGUI) and a User Application Interface (UAI) tailored to Brain Computer Interface (BCI) interaction, designed for people with severe communication needs. The IGUI has three components; a two way interface for communication with BCI2000 concerning user events and event handling; an interface to user applications concerning the passing of user commands and associated device identifiers, and the receiving of notification of device status; and an interface to an extensible mark-up language (xml) file containing menu content definitions. The interface has achieved control of domotic applications. The architecture however permits control of more complex 'smart' environments and could be extended further for entertainment by interacting with media devices. Using components of the electroencephalogram (EEG) to mediate expression is also technically possible, but is much more speculative, and without proven efficacy. The IGUI-BCI approach described could potentially find wider use in the augmentation of the general population, to provide alternative computer interaction, an additional control channel and experimental leisure activities.

## General Terms

Experimentation

## Keywords

Brain Computer Interfaces, user interface, domotic control, entertainment.

## 1. INTRODUCTION

Degenerative diseases or accidents can leave a person paralyzed yet with full mentally function. There has been significant research into creating brain mediated computer control [1] and assistive equipment that can be controlled by the brain, such as a wheelchair mounted robotic arm system [2].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Augmented Human Conference, April 2-3, 2010, Megève, France.  
Copyright © 2010 ACM 978-1-60558-825-4/10/04...\$10.00.

A Brain-Computer Interface (BCI) may be defined as a system that should translate a subject's intent (thoughts) into a technical control signal without resorting to the classical neuromuscular communication channels [3]. The key components are signal acquisition to acquire the electroencephalogram (EEG), signal processing to extract relevant features and translation software to provide appropriate commands to an application. Applications include computer and environmental control, but entertainment applications are also under investigation. It is of course possible that the application could provide some opportunity for self expression and creativity. Figure 1 illustrates some possibilities of BCI for augmenting the human: listening to music, controlling photographs, watching films, or influencing music or visual arts.

The ability to apply a BCI to the control of multiple devices has previously been explored [4]. It has also been demonstrated that BCI technology can be applied to many assistive technologies: - wheelchair control [4],[5],[6] computer speller [7],[8],[4], web-browser [9], environment control [10],[11],[12] and computer games [13],[14]. Smart homes technology is also an active area of research [15],[16], both for assistive applications and to enhance 'lifestyle'. This BCI paradigm offers the opportunity for automated control of domotic devices and sensor interaction, for the purpose of providing an integrated ambient assistive living space.

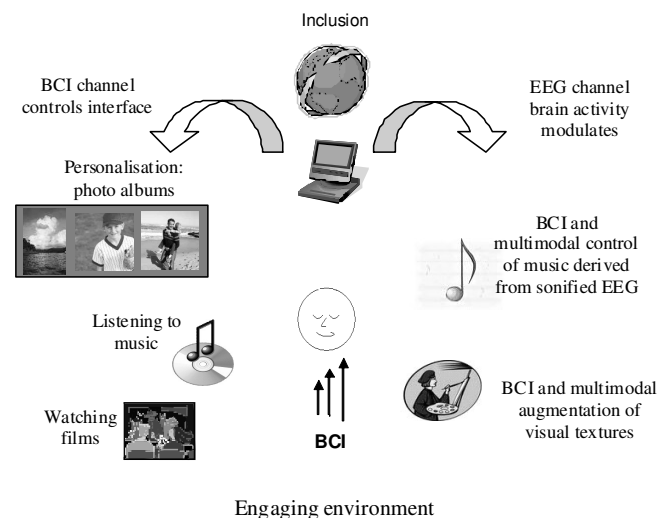


Figure 1: Uses of BCI for inclusion and augmentation of disadvantaged citizens

The merging of these complex and emerging technologies is not a trivial issue. The European Union has addressed this with BRAIN (FP7-2007-ICT-2-7.2), and other BCI projects, which come under the “Future BNCI” umbrella [17]. BRAIN is dedicated to providing a solution for controlling multiple domotic devices. The project adopts a multi-disciplinary approach to address issues ranging from improving BCI interaction [18],[19]; to integrating smart homes solutions. BRAIN’s focus concerns the application architecture and modes of user interaction. It offers an application interface with a wrapper for integrating domotic standards and protocols, if required, known as the universal application interface (UAI). An intuitive graphical user interface (IGUI) provides an on screen menu structure which interacts with the UAI to achieve device control. The IGUI also interacts with BCI components, currently implementing a high-frequency steady state visual evoked potential (SSVEP) BCI paradigm.

## 2. ARCHITECTURE DESIGN

Blankertz [7] comments that a ‘major challenge in BCI research is intelligent front-end design’. To be effective a BCI used within a domestic setting has to interact with multiple devices and software packages. Within BRAIN we have designed an architecture that accommodates modification, facilitates substitution of existing components and to which functionality can be added. This facilitates emerging domotic devices or upgrades in BCI technologies. The architecture is modular in approach with each component being highly cohesive and exhibiting loose coupling to other modules each with clearly defined interfaces. The human interface is intuitive, to unify principles of operation across devices thereby reducing learning and overhead of operation. This is crucial with BCI, which suffers very slow communication rates, with user invoked error choices posing a major problem. The architecture of the application and the interface must also be capable of supporting context aware technologies such as predictive commands and command sets based upon models of previous interactions.

The concept of a separate application working in conjunction with BCI technologies was proposed by Mason et al [20]. This conforms to the end user elements of the controller interface and assistive devices as defined in the framework. It is also keeping with the design philosophy of the BCI2000 [21]. The BRAIN architecture incorporates the flexible approach to the brain-computer interface of the BCI2000 general purpose platform. The platform is designed to incorporate signals, signal processing methods, output devices and operating protocols. The BCI2000 interface is minimal, consisting of Universal Data Packets (UDP) the content of which can be specified in accordance with the signal processing being performed. This approach employs the packets which originate from signal processing activity [19]. Three main types of command interaction sequence have been identified: binary, analogue and hybrid, see Figure 2. Binary command interaction concerns the issuing of a single command to the UAI in a single instance. This is used to navigate menu structures, and issue single autonomous instructions. Analogue command interactions

concern the issuing of temporally continuous and progressive commands for the purpose of adjusting environmental controls, e.g. to increase volume on a media device; these interactions are vulnerable to latency. A hybrid command sequence achieves the intent of an analogue command sequence but is implemented using a binary style of interaction. It offers pre-defined options: high, medium, low; and is enacted via a single command. It is not suitable for fine-tuning.

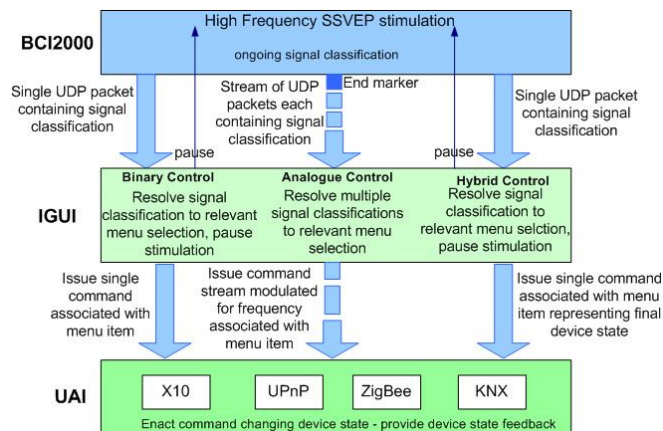


Figure 2: Operation of Binary, Analogue and Hybrid Command Sequences

### 2.1 Intuitive graphical user interface (IGUI)

An intuitive graphical user interface (IGUI) provides on screen menu display of application content. This interface is suited to operate in conjunction with BCI peripherals, transducers and the control interface. Furthermore, conforming to specified user requirements the interface is defined in such a way as to be suited to operation under various BCI paradigms whilst maintaining common principles of operation. It is applicable to all devices in the device controller module, i.e. universal application interface (UAI), or which may be envisaged at a future point. It is capable of updating content as the state or availability of the applications or devices changes over time. The user interface is also capable of handling modifications in display or operation according to user defined preferences. It is anticipated that modules will be added to the interface architecture which will provide predictive capabilities with respect to user menu selections based upon context and passed choices. The UAI acts as wrapper for multiple device interaction standards and protocols; it provides a single control interface to the IGUI, hiding the complexity of interaction. It is expected that the UAI will facilitate a variety of protocols and standards with regard to domotic devices and that the number of available devices will expand.

The IGUI has three major interfaces. The first is a two way interface for communication with BCI2000 concerning user events and event handling. The second, a two way interface to the UAI concerning the passing of user commands and associated device identifiers, and the receiving of notification of device status. The third is an interface to an extensible mark-up language (xml) file containing menu content definitions. This file although initially defined ‘off-line’ is thereafter written to by the UAI and read by the IGUI, dynamically. Should the need

arise substitution of packages is possible provided that the interface definitions are adhered to or an additional interface wrapper is implemented for the purpose of ensuring interface compatibility. In this manner, should the need arise; the IGUI can support an alternative signal processing mechanism to BCI2000, and has already been interfaced to an 'OpenBCI' platform [18]. The IGUI can potentially support other forms of device interaction instead of the UAI, for instance: a dedicated navigational application for a wheelchair, with distributed control, executing macro commands and feeding back state information. The IGUI can support substitution of menu content by changing the definitions in the xml file.

The following sequence represents IGUI-UAI device interaction

1. BCI2000 raises user commands to the IGUI via an incoming data packet. The command is given significance based upon the context of the user interface which is obtained from the xml menu file.
2. The specific command and associated device identification is passed to the UAI which handles commands according to the appropriate protocol or standard and instigates actual device interaction.
3. The status of devices as they join or leave the smart homes network is updated via the xml menu definition file; where devices are either enabled or disabled as appropriate. The IGUI is informed as device status is modified so that the menu can be re-parsed and the display updated accordingly.
4. For the purpose of receiving incoming messages the IGUI implements two listening threads. One dedicated to listening for incoming BCI2000 data packets – on thread `UDPListener` and one dedicated to listening for incoming UAI redisplay events and unpackaged BCI2000 communications on thread `EventQueueListener`. Clearly, it does not make sense to allow the user to issue a command as the menu display is being up-dated. The device that the user may wish to interact with may no longer be available; neither does it make sense for the menu to be redisplayed at the same time a user command is being processed as the outcome may affect the menu display. For this reason mediation has to take place between events raised on either thread and each event is processed sequentially.

Interaction with BCI2000 is based upon the reception and sending of data packets. The Internet Protocol (IP) address and communication port of a computer supporting BCI2000 is known to the IGUI. Using these, a thread is initiated for the purpose of listening for incoming packets. On packet reception, the data is unpacked and the nature of the incoming user command determined. The appropriate message is placed on the `EventQueue`. The UAI may also write an event to the `EventQueue`, essentially the UAI indicates when and how the menu should be re-parsed and redisplayed. The IGUI `EventQueueListener` monitors the `EventQueue` length. If an event is detected in the queue the IGUI reads the event, instigates the appropriate processing and removes the event from the queue.

User commands passed through BCI2000 can either affect the menu items displayed or can initiate the issuing of a device command through a call to the UAI. In either case, a data packet is sent to BCI2000 indicating that LED operation should be suspended. The appropriate processing is performed and a further data packet is sent to BCI2000 re-initiating LED operation. Where the user has indicated the issuing of a device command, the appropriate device identifier is read from the xml menu definition file with the associated command. A command notification is raised against the UAI using these two parameters. The UAI returns a device status indicator to the IGUI.

The purpose of the interface is to offer the ability to manipulate menu context and issue simple instructions as part of an ongoing sequence of communication. BCI is a low bandwidth form of communication; the interface must offer the maximum amount of control for the minimum amount of interaction. The interface must optimise engagement for the user, giving them a sense of grounding with the application domain, offering a pathway towards task completion and giving a sense of accomplishment and progression as each step in a sequence of actions is achieved.

The IGUI and the UAI share access to a common menu structure. This menu is implemented in static xml with a separate parsing module. The structure as implemented is hierarchical, however, for future implementations, it is possible to declare traversal paths in-order to provide short-cuts, for instance return to a high level menu on completion of a sequence of tasks. The current menu details several locations: back garden, two bedrooms, bathroom, dining room, front garden, hall, kitchen, living room. Devices are grouped according to each room. Where devices are in a communal area, the user's menu declaration lists the communal room and the device. A user's menu declaration will not normally list menu items which are only of significance to another user. When the UAI detects that a device is available to the network, the device status on the menu declaration will be updated to 'enabled'. Provision within the xml declaration has also been made such that, should a device be judged to be sufficiently significant it can be made constantly available, through the use of a 'sticky' status to indicate permanent on screen display. It is also possible to use a non location based groupings such as 'general' to collect devices and applications together which do not have a single location, for example spellers and photo albums. Should the interface be used for some other purpose it is possible to implement a different classification mechanism, for instance grouping by functionality, or if necessary no classification mechanism. This is done by simply using replacing the xml declaration. Where devices or locations are to be added, the xml declaration can be expanded accordingly.

The sample xml declaration (below) lists two forms of item. 'Node' items have sub-item declarations (e.g., `Bedroom1`). 'Leaf' items are used to associate a specific physical device or software package to a device/package interface command, e.g. `x10Light1`. All menu items have an associated graphical display. The location of the graphics file is declared in the icon tag of the menu item in the xml declaration. Currently the menu implementation uses static xml. Provision has been made in the

IGUI interface for the passing of an object containing a similar xml declaration for dynamic content. Dynamic content of the same format can be parsed and displayed using existing mechanisms. Dynamic xml is relevant where content may be subject to frequent change, such as listing available files on a media server (e.g. movie titles).

```

<menu_list_item>
<label>Bedroom1</label>
<enabled>True</enabled>
<sticky>False</sticky>
<icon>Bedroom1.jpg</icon>
<on_selection>
  <menu_list_item>
    <label>Lighting</label>
    <device_Id>x10Light1</device_Id>
    <enabled>True</enabled>
    <sticky>False</sticky>
    <icon>Bedroom1/Lighting.jpg</icon>

    <on_selection>
      <command>BinaryLight.
        toggle_power</command>
    </on_selection>

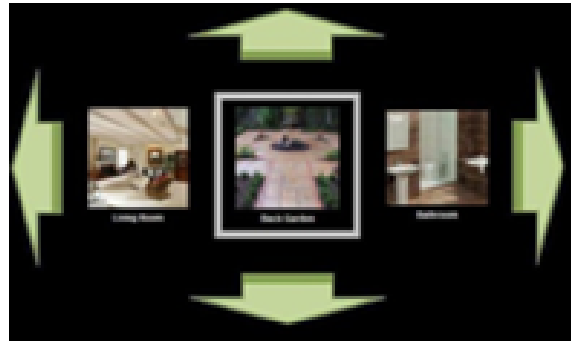
  </menu_list_item>
</on_selection>
</menu_list_item>

```

The xml menu declaration represents menu content. The user needs a mechanism for manipulating content in as an effective manner as possible, in order to traverse the menu hierarchy and to pass correctly formulated commands to the UAI. Currently the supported BCI paradigm implements high-frequency SSVEP as a mechanism for user interaction, but it is anticipated that other BCI paradigms ('oddball' stimulus and intended movement) will be supported over time. Studies have reported that up to 48 LEDs can be used. These operated between 6-15Hz at increments of 0.195Hz [22]. However, making this many signals available to the user in a meaningful manner using a conventional screen interface requires a degree of mapping which may be beyond both the interface and beyond the user's capabilities and inclinations. Similarly, many devices (cameras, mp3 players, printers) with restricted input capabilities use a four-way command mapping as an interface of choice. Using such a command interface it is possible to cycle through lists of menu items (left/right), to select commands or further menu option (down), and to reverse selections or exit the system (up). Using less than four commands produces an exponential command burden upon the user as cycle commands (left/right) increase and selection commands (down) cannot be applied in a single action. It was decided that a four-way command interface would be optimal. The LEDs are placed at the periphery of the screen with command icons central to the display [23].

The command interface Figure 3, displays the icons relating to three menu items central to the screen. The central icon represents the current menu item; as such it is highlighted using a lighter coloured frame. Icons to either side, provide list orientation to the user, to suggest progression and to suggest alternative options. Under the current SSEVP paradigm the four

command arrows presented on the screen point to four peripheral LEDs. For the purpose of interface testing and to provide a potential support facility for the carers of users, the four command arrows can be activated using a standard mouse. Under different BCI paradigms arrows would still be present on the screen but they would function in a slightly different manner. Under P300 it is anticipated that an additional module would govern the time sequenced animation of the arrows, thereby providing a synchronised stimulus for the 'oddball' effect. Alternatively voluntary responses can be used to control cursor movement towards arrows (ERD/ERS intended movement paradigm).



**Figure 3: Intuitive Graphical Command Interface for BRAIN Project Application**

The screen displays location level menu items. At this menu level the icons are photographs which in the real world context could relate to the users own home. The use of photographs and real world images are intended to make the interface use more intuitive to the user and to reduce the cognitive load of interacting with the menu structure. At a lower menu level general concept images have been used, albeit still in a photographic format. Specifically, individual lights are not represented; instead a universal image of a light bulb is used. It is felt that at this menu level the user will already have grasped the intent of the interface. Furthermore, the concept of device interaction at the command level is made universal by this approach, such as a tangibly visible interaction of turning a light on and an invisible interaction such as turning the volume of a device up. Once again the flexibility of the application is demonstrated. The intuitive feel of the interface can be modified by simply replacing the graphics files. For instance, a 'younger' look and feel can be obtained by replacing photographic representations with cartoon style drawings. On screen display of icons are supported by associated labels, these are represented by tags in the xml menu declaration. The labels are used to make the meaning explicit; however the interface has been devised in such a way as to ensure that literacy is not required.

### 3. Application Interface

Smart Homes are environments facilitated with technology that act in a protective and proactive function to assist an inhabitant in managing their daily lives specific to their individual needs. Smart homes technology has been predominately applied to

assist with monitoring vulnerable groups such as people with early stage dementia,[24] and older people in general [25] by optimising the environment for safety. The link between BCI and Smart homes is obvious, as it provides a way to interact with the environment using direct brain control of actuators. Our contribution uses a software engineering approach, building an architecture which connects the BCI channel to the standard interfaces used in Smart Homes so that control, when established, can be far reaching and tuned to the needs of the individual, be it for entertainment, assistive devices or environmental control. Thus a link to the standards and protocols used in Smart Home implementations is important.

A BCI-Smart Home channel could allow users to realize several common tasks. For instance, to switch lights or devices on/off, adjust thermostats, raise/lower blinds, open/close doors and windows. Video-cameras could be used to identify a caller at the front door, and to grant access, if appropriate. The same functions achieved with a remote control could be realized (i.e. for a television, control the volume, change the channel, 'mute' function). In a media system, the user could play desired music tracks.

### 3.1 Standards and Protocols

While the underlying transmission media and protocols are largely unimportant from a BCI user perspective, the number of standards provides an interoperability challenge for the software engineer. Open standards are preferred. A number of standards bodies are involved; the main authorities are Institute of Electrical and Electronics Engineers (IEEE), International Telecommunication Union (ITU-home networking) and International Standards Organisation (ISO). Industry provides additional de-facto standards. Given the slow 'user channel', BCI interaction with the control aspects of domotic networks requires high reliability with available bit-rate transmission, being of much lesser importance.

Domotic standards for home automation are based on either wired or wireless transmission. Wired is the preferred mode for 'new' build Smart Homes, where an information network may be installed as a 'service' similar to electricity or mains water supply. Wireless networks can be used to retrofit existing buildings, are more flexible, but are more prone to domestic interference, overlap and 'black spots', where communication is not possible. Wireless networks normally use work using radio frequency (RF) transmission and can use Industrial Scientific and Medicine (ISM) frequencies (2.4GHz band) or proprietary frequencies and protocols. Infra-red uses higher frequencies which are short range and travel in straight lines (e.g. remote control for television control).

The Universal Plug and Play (UPnP) architecture offers pervasive peer-to-peer network connectivity of PCs, intelligent appliances, and wireless devices. UPnP is a distributed, open networking architecture that uses TCP/IP and HTTP protocols to enable seamless proximity networking in addition to control and data transfer among networked devices in the home. UPnP does not specify or constrain the design of an API for applications running on control points. A web browser may be used to control a device interface. UPnP provides interoperable

specification, offering the possibility of 'wrapping' other technologies (e.g. where a device is not UPnP compliant). UPnP enables data communication between any two devices under the command of any control device on the network. UPnP technology can run on any medium (category 3 twisted pairs, power lines (PLC), Ethernet, Infra-red (IrDA), Wi-Fi, Bluetooth). No device drivers are used; common protocols are used instead.

The UPnP architecture supports zero-configuration, invisible networking and automatic discovery, whereby a device can dynamically join a network, obtain an IP address, announce its name, convey its capabilities upon request, and learn about the presence and capabilities of other devices. Dynamic Host Configuration Program (DHCP) and Domain Name servers (DNS) are optional and are only used if they are available on the network. A device can leave a network smoothly and automatically without leaving any unwanted state information behind. UPnP networking is based upon IP addressing. Each device has a DHCP client and searches for a server when the device is first connected to the network. If no DHCP server is available, that is, the network is unmanaged, the device assigns itself an address. If during the DHCP transaction, the device obtains a domain name, the device should use that name in subsequent network operations; otherwise, the device should use its IP address.

Open Source Gateway Interface (OSGi) is middleware for the Java platform. OSGi technology provides a service-oriented, component-based environment for developers and offers a standardized ways to manage the software lifecycle. The OSGi platform allows building applications from components. Two (or more components) can interact through interfaces explicitly declared in configuration files (in xml). In this way, OSGi is an enabler of expanded modular development at runtime. Where modules exist in a distributed environment (over a network), web services may be used for implementation. The OSGi UPnP Service maps devices on a UPnP network to the Service Registry.

### 3.2 Interoperability with existing smart home interface

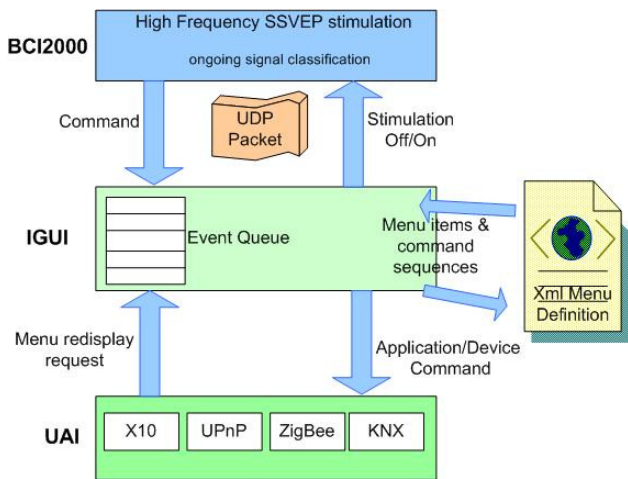
It is important that the architecture developed can interoperate with existing and future assistive technology. A BRAIN partner, The Cedar Foundation, has sheltered apartments (Belfast) which are enabled for non-BCI Smart Home control. Each apartment is fully networked with the European Installation Bus (EIB) for home and building automation [26]. Into this, peripherals are connected which can be operated via infra-red remote control [27]. These peripherals when activated carry tasks that tenants are not physically able to perform. Examples include door access, window and blind control, heating and lighting control and access to entertainment. Whilst this was 'state of the art' technology at the time of development, KNX has replaced EIB as the choice for open standard connectivity. This reinforces the need for interoperability within a modular architecture, if BCI is to be introduced to the existing configuration.



### 3.3 A Universal Application Interface

The Universal Application Interface (UAI) aims to interconnect heterogeneous devices from different technologies integrated in the home network, and to provide common controlling interface for the rest of the system layers. Figure 4 illustrates how the interfaces between BCI2000, IGUI, menu definition and UAI.

UAI control is based on UPnP specification, which provides protocols for addressing and notifying, and provides transparency to high level programming interfaces. The UAI maps requests to events, generates the response to the user's interaction, and advertises applications according device. The UAI infers the device type and services during the discovery process, including the non-UPnP devices, which can be wrapped as UPnP devices with the automatic deployment of device proxies.



**Figure 4: Interaction of the Intuitive Graphical Command Interface and Universal Application Interface**

The UAI is divided into three modules: 1) *Devices Control Module* which interacts directly with the UPnP devices. It consists of a Discovery Point and several Control Points for the different types of UPnP services. It also includes the UPnP wrappers needed to access the non UPnP devices. 2) *Context Aware Module*, which is capable of triggering automatic actions when certain conditions are met without the need of user intervention. The module receives events from the applications and devices and invokes the actions resulting from the evaluation of a set of predefined rules. 3) *Applications Layer*, which provides the interactive services the user will control through the BCI. Applications include domotic control, which allows the user to control simple devices; and entertainment, which allows the user to control a multimedia server, e.g. to watch movies.

The purpose of the UAI is to provide a uniform platform for device inter-action based upon masking the complexity of numerous domotic device standards and communication protocols. Flexibility and robustness in design is evidenced by the fact that it is possible to substitute the two core components or for the core components to interact with other hardware and software configurations, if necessary. For instance by simply

replacing a communication wrapper the IGUI could interface with a different BCI package, or by replacing the UAI the IGUI could be harnessed for other control purposes, for example driving a robot. It is also possible for the IGUI to be substituted and for a different command interface to call the services of the UAI.

The UAI is also flexible, additional standards can be added without modifying the core command processing and device handling modules. By incorporating new standards it is possible to interact with an increasing number of devices without radically modifying other aspects of the application device architecture. By presenting an architecture which facilitates the up-grading of existing standards it is also possible to interact with existing devices in a more efficient manner.

## 4. BCI FOR CREATIVE EXPRESSION

The link between EEG and performance has been established by Miranda and Brouse [28], who used a BCI to play notes on an electric piano. The piano didn't play specific songs as such, but more rhythms of similar notes put together. There was not much choice in what the application could do but it created music using purely the power of the mind. The system was difficult to use but provided an outlet for disabled users. Even if not particularly creative, music and art therapy can be used to improve quality of life for severely disabled individuals.

Within the area of EEG there has been a level of interest in sonification of the different frequency bands within the waveforms. For some, the objective has been to create an "auditory display" of the waveforms as a method to portray information. For example a sonic representation of the heart rate can be a powerful mechanism for conveying important information, complementing the visualization of the data. Biosignals such as EEG and electromyogram (EMG) or muscle activity can also be used to generate music. Using EEG as a forum for musical expression was first demonstrated in 1965 by the composer Alvin Lucier through his recital named "Music for Solo Performer" [29]. Here manipulation of alpha waves was utilized to resonate percussion instruments. Rosenboom [30] used biofeedback as a method of artistic exploration by composers and performers.

There has been recent work within BCI applications for music creation by researchers Miranda, Arslan, Brouse, Knapp and Filatriau. Brouse and Miranda [31] developed the eNTERFACE (<http://www.enterface.net/>) initiative resulting in two types of instrument. The first was referred to as the BCI-Piano and used a music generator driven by the parameters of the EEG. The second instrument was the InterHarmonium and enabled geographically separated performers to combine the sonification of their EEG in real time. In addition to generating music, EEG synthesis of visual textures have also been investigated [32].

The importance of enabling a level of creativity and self-expression to be given to highly physically disabled people through the use of EEG and bio-signal sonification is highlighted by the Drake music project [33]. Assistive technology is used, enabling compositions by musicians, with

only a limited level of motor movement. By incorporating BCI within such a framework would expand the ability of self-expression to people with much more severe physical disabilities. Multi-modal interaction, allowing EMG for example to augment the EEG signal, is the basis of the BioMuse system. In addition the combination of enabling the generation of both music and visual display as demonstrated by Arslan et al [34] creates enhanced opportunity for self-expression.

## 5. CONCLUSIONS

We describe an intelligent graphical user interface (IGUI) tailored to Brain Computer Interface (BCI) interaction, designed for people with severe communication needs. The interface has achieved control of simple domestic applications, via a user applications interface (UAI). The architecture described however permits control of more complex 'smart' environments and will be extended further for entertainment by interacting with media devices. While the use and efficacy of BCI for creative expression is still highly speculative, the technical approach adopted with the IGUI can be easily adapted to the generation of relevant features from the EEG. All that is required is agreement upon syntax and content of UDP packets and additional signal processing. The IGUI approach described could potentially find wider use in the general population, to provide alternative computer interaction, an additional control channel and experimental leisure activities.

## 6. ACKNOWLEDGMENTS

The BRAIN consortium gratefully acknowledge the support of the European Commission's ICT for Inclusion Unit, under grant agreement No. 224156.

## 7. REFERENCES

- [1] Wolpaw, J. R., McFarland D. J. Control of a two-dimensional movement signal by a non-invasive brain-computer interface in humans, Published by Proceedings of the National Academy of Sciences of the United States of America, December 21, 2004 vol. 101 no. 51
- [2] Laurentis, K.J., Arbel Y., Dubey R., Donchin E. Implementation of a P-300 brain computer interface for the control of a wheelchair mounted robotic arm system. Published by ASME in the Proceedings of the ASME 2008 Summer Bioengineering Conference (SBC2008), June 25-29, pp. 1-2 (2008).
- [3] Blanchard, Gilles, Blankertz, Benjamin, (2003) BCI Competition 2003—Data Set IIa: Spatial Patterns of Self-Controlled Brain Rhythm Modulations, Published by IEEE.
- [4] Millan J.R, 'Adaptive Brain Interfaces', Communications of the ACM, March 2003, Vol 46 No 3, pp75-80.
- [5] Valsan G, Grychtol B, Lakany H, Conway B.A, 'The Strathclyde Brain Computer Interface', IEEE 31st. Annual International Conference on Engineering in Medicine and Biology Society, Sept 2009, pp606-609.
- [6] Leeb R, Friedman D, Müller-Putz G.R, Scherer R, Slater M, Pfurtscheller G, 'Self-Paced (Asynchronous) BCI Control of a Wheelchair in Virtual Environments: A Case Study with a Tetraplegic', Journal of Computational Intelligence and Neuroscience, Vol 2007.
- [7] Blankertz B, Krauledat M, Dornhege G, Williamson J, Murray-Smith, Müller K.R, 'A Note on Brain Actuated Spelling with the Berlin Brain-Computer Interface', Lecture Notes on Computer Science, 2007, Vol 4555, pp759-768, Heidelberg, Springer.
- [8] Felton E, Lewis N.L, Willis S.A, Radwin R.G, 'Neural Signal Nased Control of the Dasher Writing System', IEEE 3rd. International EMBS Conference on Neural Engineering, May 2007, pp366-370.
- [9] Bensch M, Karim A.A, Mellinger J, Hinterberger T, Tangermann M, Bogdan M, Rosenstiel W, Birbaumer N, 'Nessi: An EEG-Controlled Web Browser for Severely Paralyzed Patients', Journal of Computational Intelligence and Neuroscience, Vol 2007.
- [10] Piccini L, Parini S, Maggi L, Andreoni G, 'A Wearable Home BCI System: Preliminary Results with SSVEP Protocol', Proceedings of the IEEE 27th. Annual Conference Engineering in Medicine and Biology, 2005, pp5384-5387.
- [11] Teo E, Huang A, Lian Y, Guan C, Li Y, Zhang H, 'Media Communication Centre Using Brain Computer Interface', Proceedings of the IEEE 28th. Annual Conference Engineering in Medicine and Biology, 2006, pp2954-2957.
- [12] Bayliss J, 'Use of the Evoked Potential P3 Component for Control in a Virtual Apartment', IEEE Transactions on Neural Systems and Rehabilitation Engineering Vol 11, No 2, 2003, pp113-116.
- [13] Martinez P, Bakardjian H, Cichocki A, 'Fully Online Multicommand Brain-Computer Interface with Visual Neurofeedback Using SSVEP Paradigm', Journal of Computational Intelligence and Neuroscience, Vol 2007.
- [14] Berlin Brain-Computer Interface – The HCI Communication Channel for Discovery', International Journal Human-Computer Studies, Vol 65, 2007 pp460-477.
- [15] Chan M, Estève D, Escriba C, Campo E, 'A Review of Smart Homes – Present State and Future Challenges', Computer Methods and Programs in Biomedicine, Vol 91, 2008, pp55-81.
- [16] Poland, M.P, Nugent, C.D, Wang, H, Chen L, 'Smart Home Research: Projects and Issues', International Journal of Ambient Computing and Intelligence, Vol 1, No 4, 2009, pp32-45.
- [17] Progress in Brain/Neuronal Computer Interaction (BNCI), <http://hmi.ewi.utwente.nl/future-bnci>, accessed Jan 2010.
- [18] Durka P, Kus R, Zygierevicz J, Milanowski P and Garcia G, 'High-frequency SSVEP responses parametrized by multichannel matching pursuit', Frontiers in Neuroinformatics. Conference Abstract: 2nd INCF Congress of Neuroinformatics. 2009.
- [19] Garcia G, Ibanez D, Mihajlovic V, Chestakov D, 'Detection of High Frequency Steady State Visual Evoked

- Potentials for Brain-Computer Interfaces', 17th European Signal Processing Conference, 2009.
- [20] Mason S.G, Moore Jackson M.M, Birch G.E, 'A General Framework for Characterizing Studies of Brain Interface Technology', *Annals of Biomedical Engineering*, Vol 33, No 11, November February 2005, pp1653-1670.
- [21] Schalk G, McFarland D, Hinterberger T, Birbaumer N, Wolpaw J.R, 'BCI2000: A General-Purpose Brain-Computer Interface (BCI) System', *IEEE Transactions on Biomedical Engineering* Vol 51, No 6, June 2004, pp1034-1043.
- [22] Gao X, Xu D, Cheng M, Gao S, 'A BCI-Based Environmental Controller for the Motion-Disabled', *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol 11, No 2 June 2003, pp137-140.
- [23] Parini S, Maggi L, Turconi A Robust and Self-Paced BCI System Based on a Four Class SSVEP Paradigm: Algorithms and Protocols for a High-Transfer-Rate Direct Brain Communication', *Journal of Computational Intelligence and Neuroscience*, 2009.
- [24] ENABLE project (2008). Can Technology help people with Dementia? Retrieved July 7, from <http://www.enableproject.org/>
- [25] Kidd, C.D., Orr, R.J., Abowd, G.D., Atkeson, C.G., Essa, I.A., MacIntyre, B., Mynatt, E., Starner, T.E., Newstetter, W. (1999). The Aware home: a living laboratory for ubiquitous computing research, *Proceedings of 2nd International workshop on cooperative buildings Integrating Information, Organization, and Architecture*, 191 - 198 .
- [26] Kolger, M. 2006. Free Development Environment for bus coupling units of the European Installation Bus. Available at <http://www.auto.tuwien.ac.at/~mkoegler/eib/sdkdoc-0.0.2.pdf> . [Accessed on 21.07.09]
- [27] SICARE SENIOR PILOT, 2008. Available at: <http://www.reflectivepractices.co.uk/cms/index.php?option=displaypage&Itemid=50&op=page&SubMenu=> [Accessed on 21.07.09].
- [28] Miranda, E. R. and Brouse, A. (2005a). Interfacing the Brain Directly with Musical Systems: On developing systems for making music with brain signals, *Leonardo*, 38(4):331-336
- [29] Lucier, A (1965). Music for the solo performer, <http://www.emfinstitute.emf.org/exhibits/luciersolo.html> last accessed July 2007.
- [30] Rosenboom, D. (2003). Propositional Music from Extended Musical Interface with the Human Nervous System, *Annals of the New York Academy of Sciences* 999, pp 263.
- [31] Brouse, A., Filatriau, J.-J., Gaitanis, K., Lehembre, R., Macq, B., Miranda, E. and Zenon, A. (2006). "An instrument of sound and visual creation driven by biological signals". *Proceedings of ENTERFACE06, Dubrovnik (Croatia)*. (Not peer-reviewed report.)
- [32] Filatriau, J.J., Lehembre, R., Macq, B., Brouse, A. and Miranda, E. R. (2007). From EEG signals to a world of sound and visual textures. (Submitted to ICASSP07 conference).
- [33] Drake music project (2007), <http://www.drakemusicproject.org/>, last accessed July 2007
- [34] Arslan, B., Brouse, A., Castet, J., Lehembre, R., Simon, C., Filatriau, JJ and Noirhomme, Q. (2006). A Real Time Music Synthesis Environment Driven with Biological Signals, *IEEE International Conference on Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings*, vol. 2.

---

<sup>i</sup> Additional authors: M.D..Mulvenna, H.G..McAllister, C.D..Nugent, Faculty of Computing and Engineering, University of Ulster, Shore Road, Jordanstown, Co. Antrim BT37 0QB, UK