

General Questions	Institution Person	LAP EPFL Edoardo Charbon	distlab Diku Mads Dydensborg	NTT CoCoMo Anthony Tarlano	PCCV ETHZ Stavrod Antifakos
What is your main application/use for the BNodes?		Local Positioning System (LPS) Com for Virtual Human Interface optical devices	Experiments with TinyOS and Bluetooth.	Ubiquitous computing using sensors	- sensor data streaming from multiple nodes - sensor data preprocessing and sending results from multiple nodes.
How many applications/demos have you implemented on the Bnodes? And with how many nodes?		LPS	A number of test applications, one major application, on 5 nodes.	None currently, we have only evaluated the the device. We expect to be using the more frequently in the near future	- avalanche demo - proactive furniture (small demo - proactive furniture experimental sensor data streaming (diploma thesis)
How fast did you get your first application running? How much effort was this?		The very first, one morning. The real LPS took about 15 days, including gaining familiarity with the propagation/multi-path models	Our first application was porting TinyOS to the BNode, and it took several months, and was a lot of effort. I do not have exact numbers.	N/A	once all the hardware was ready documentation was fine to get the first apps running
How many devices would you like to deploy until Q4/2004? At what pricerange?		I would need a dozen more. I'd like the price to be at most CHF100. CHF20 would be great!	I have no idea. Perhaps, if they were in the < 20 euro range, about 30 pcs.	20-30 at the lowest price possible	depends on applications addressed, maximum another 20
Do you see yourself as a user of Bnode technology or as a core developer? What are your main interests in this platform?		core developers as in LPS we could only use RSSI to measure distance. It was not possible to gain access to inside signals, i.e. Before demodulation.	User. Main interest is a hardware platform for experimenting with Bluetooth on a tiny system.	My main interests are in proof of concepts.	and computation power. 1.) sensor data streaming (high bandwidth) over multiple nodes to collect data for off line processing, 2.) demo development using multiple nodes. we do use something else, we also use the smart-its communication platform and sensor board. depending on what is better for the application
Why do you use BNodes and not something else?		Ease of use, cost, availability	It was available.	personal contact	software, we have also implemented a i2c connection to the smart-its sensor board. software is available at: http://www.vision.ethz.ch/antifako/ -> perception api (i2c version)
Have you been using the BNode software? Or something else? Have you contributed to the BNode software with bugfixes or enhancements?		we just used the standard code and built app code	ported/used TinyOS (implementation part of the TinyOS CVS rep in contrib/tinybt). We have not contributed to the bnode tree.	Not yet	
What is your experience with the architecture of the BNode software? Is it an appropriate approach, or are there any real flaws or shortcomings?		I am not sure I am qualified to comment. You should ask Michal.	It seems appropriate.	Good approach	event based architecture makes sense for most applications It is not always such a reliable platform. connections are often closed without error messages when using multiple nodes and streaming sensor data.
Is it a reliable platform? Are you seeing stability problems or inexplicable errors that are real showstoppers?		No prob on our side that I can remember, but Michal should be more specific	We have not suspected any hardware problems. We have a faulty node, but I have no idea why it is faulty.	N/A	
Have you used any of the low-power features on the BNodes?		no	Only the sleep mode of the CPU. It would be good to be able to "somehow" have the CPU power almost completely down, and have it wake up, on activity on the serial port. I suspect this is not possible?	N/A	no
What are the main drawbacks you see in this platform? What should be improved?		2.) unpredictability of antenna gain. Perhaps a discrete antenna would be preferable 3.) lack of flexibility in using other		N/A	Definitely, reliability + data throughput. Time should be invested into testing and development of system software
Would you be interested in having					
Multiple radio frontends?		yes, with physical interchangeability or chip in place	It would be nice to be able to use a non-bluetooth radio (fixed frequency)	yes	maybe, not quite shure.
A low power/full custom (Chipcon,RFM,Wisenet) type radio (instead of or additional to Bluetooth)?		yes	yes	yes	maybe usefull for data streaming
More/less memory?		no	We are still below the 4 KB mark on most applications.	more	if, then more.
More/less interfaces?		no	no	more	sensor and actuators, i2c is always required, pwm (pulse width modulated signals) going out in some form is always important for connecting actuators.
More/less processing capabilities?		no	no	more	is okay,
A smaller/fewer resource platform?		no	It would be nice if the form factor could be reduced.	no	no, prefer to have more resources than less.
A standard power supply/battery?		yes, with a single AAA if possible	no	yes	yes, would take some work off the application developer.
Your personal wishlist for BNode Rev 3		1.) be part of definition of future multi-standard development 2.) access to internal FE signals (before demod) 3.) more docs 4.) less power - have you considered the TI chips?	Lower power consumption - sleep modes wake up on radio interface activity. Perhaps some instructions on how to build a sensor for the platform.	Cheap price, solid design, good code...	better software (more reliable) (haven't tested the newest version though.)

DSG ETHZ
Frank Siegemund

research in Ubiquitous Computing; attaching BT nodes to everyday objects to augment them with the ability to communicate with each other and collaboratively provide services to people in their everyday environment

Around 10 real applications, up to 4 nodes

At the time I started it was quite difficult, because I had to program a BTstack at first... - around 1 month

this is really arbitrary: 20, up to 250 Euro

I am programming at both, lower and higher levels; for main interests see question 1.

relatively large amount of resources; easy integration into existing environments; interoperability with handheld devices such as mobile phones and PDAs

use my own software

do not know the official release in depth

yes, 2 BTnodes have been damaged while programming and are not working anymore

I was not aware that there are low-power features other than that regarding the microcontroller

high energy consumption; lack of possibilities to control energy consumption of most electrical parts on the boards (switch off Bluetooth-module, etc.); typical drawbacks of Bluetooth: connection establishment + inquiry delay, etc.

Sounds interesting; but not if this increases size considerably

Definitely not instead of Bluetooth; otherwise one would have another version of the Berkeley motes

current memory capacity is sufficient; definitely not less memory!

more, if this does not increase size

current status is good

No.

???
interesting is not whether I want more/less interfaces, more/less memory, but the relation between number of interfaces <-> size <-> energy <-> memory, etc do any necessary programming.
F

EPFL
Urs Hunkeler

1) stationary node that will provide information to mobile nodes. The information should be updated via other stationary nodes 2) (not sure yet) Mobile node that will connect itself to stationary nodes to collect information

1 application with two fixed nodes (only bttester at this time)

Between 1 to 2 weeks. Documentation could be improved (how about a "beginner's tutorial"?). But the sample programs were really useful

500 - 1000 (maybe more) at less than 200.- SFr (less than 100.- would be ideal)

Between. I'd like not to have to worry about low-level coding. However I do develop applications (so I am not just a user)

It is a complete, standalone and small product without unnecessary features at an acceptable price

I do use BTnode software. I still have some bizarre errors and maybe I get around to investigate them. In that case I might provide bugfixes

It is an approach that is easy to understand. I did not encounter short-comings in functionality. However it is a bit difficult to determine the cause of certain errors (if I get an error number, where did that error occur? is there ambiguity, i.e. two

I do have some bizarre errors, but this does not, at the moment, cause severe problems. There is also a problem when a device leaves the transmission range and reappears later. It seems that different technologies don't have the same timeout for disconnect

Not yet, but I intend to use them

I think that a ready made, off the shelf product (including a casing and a powersupply/battery) might be interesting for research and maybe even commercial products.

I don't have a time sensitive application at the moment, but this might be interesting in the future. However the price is an important factor. If you do something like this, maybe it would be an interesting idea to offer the additional radio frontends as

If, then only in addition to bluetooth. This might be interesting if it allows for a better implementation of MANETs (instead of scatter nets). Bluetooth is important because it's becoming a standard and other devices already implement this technology an

Some of the applications I have in mind would require more than 500k of memory

One serial port is enough for me

The processing capabilities seem minimalistic but sufficient to me

Seems to be ok as is

Yes, with casing. How about a complete set with casing and a solar panel for outdoor field research?

TIK ETHZ
Jan Beutel

Ad hoc networking scenarios and fast prototyping of demo apps

about 5 apps with up to 20 nodes

The first apps were a pain since we didn't have the Bluetooth and OS framework. It took us 2-3 months to get something significant rolling out.

about 100 at under 100CHF

Core developer, but actually just want to use the platform

Core developer. Btnodes are ideal for high level networking implementations and interaction with other devices.

Yes

The BT module is way too buggy and does not support the features we want.. We need to spend a lot more time on software issues and make it more stable.

Many sw issues. Hw is pretty good, only one bug known (fixed)

yes, but not in production systems

the sw and error handling, as well as a different/better bt module

rather have less, but more generic, since they use up a lot of space

No. Processing is ok.

Smaller yes in size. But not in resources.

Yes, definitely

DSG ETHZ
Harald Vogt

I want to receive and send lots of data over Bluetooth and talk to an MP3 decoder/encoder chipi

None yet

Pretty fast, thanks to good examples and tutorials

5. Price not important, should be "reasonable"

More of a user. Like to learn about embedded systems programming and communication, and ad-hoc networking.

Easy access (I am at ETH).

Used: Yes; Contributed: Not yet :-)

Seems okay.

No experience yet.

no

don't know yet

yes

Would be nice to have an additional interface.

MORE. DEFINITELY.

don't know yet

don't know

need to be smaller

I don't understand the question.

Sorry, I have to little experience yet.

DSG ETHZ Oliver Kasten	DSG ETHZ Matthias Ringwald	LCA EPFL Jacques Panchard	Vitronics Paul McCarthy	TIK ETHZ Philipp Blum
research on object tagging	prototype ubicomp applications, using bt-nodes as communicating embedded device	Collection and processing of soil and ground-water data for water management in the agricultural area.	distributed sensor, data logger	robots: one bt-node connected to every robot, bt-nodes form a piconet and relay ir-lego-messages from the local robot via bluetooth to all other robots. (prototype running, on-going work) and to
small play applications with two to four nodes	no real one. Trying to scatternets working with 15 nodes	None	1/1	one, with up to five nodes
first i had to write the bleeding bt-stack and scheduler... about a year or two ;)	very fast, but sometimes undeterministic failures to program the device	Not applicable	not yet complete	two weeks, one for the soft-uart driver, one for the lego-relay app
100 devices would be nice	having 50 in our group	50-100 nodes. Up to 5000 CHF.	1000, US\$100/each	up to 15, depending on the future of pps-mindstorms, other projects interested in lego-robotics (mics?)
core developer up to some point, then user	core developer, developing system software	Both. We want to use as many off-the-shelf components as possible, but we will do any necessary programming.	developer	user, need for some low-level modifications (soft-uart, high-resolution clock)
because of bluetooth (standard wireless interface), open source (can give away, others help developing)			open source, flexible design for bluetooth sensor node.	personal contact, hw and know-how readily available within our work group
yes, no, yes	BTnode software asynchronous events, even for communication establishment, is error-prone. For most applications, a simpler API is beneficial. The API needs to be cleaned up. E.g. using different identifiers		writing our own mostly.	yes, contributed soft-uart driver
both: it is appropriate for my work but it has some severe flaws!			not sure	yes, terminal (bt-cmd) could be more comfortable (backspace, history, ..)
there are a few stability problems that need fixing (softwarewise), hardwarewise it's fine!	It is more or less reliable. The BT Stack is unstable, and not able to recover. The BT Module crashes. This is not addresses.		no reliability problems	bluetooth stack
no	No. Indicator Reference Volume should be lower SW: Clean up API, Roktster USB Driver for Mac & Linux, Clean up implementation (e.g. duplicated code, remove code in comments)		yes.	no
software needs improvement (error model, stability), also documentation needs improvement			incomplete implementation of bluetooth stack. Board could be a bit smaller and use the more popular CSR based radios	
don't need that	yes		yes, especially the option to use 300/900MHz range radio to interface with Mica nodes	yes! power down of bluetooth
must keep bluetooth	Yes - in addition to BT		yes, see previous response	low-power, simple/no mac -> complementary to bluetooth, deterministic message delay
memory is fine, especially with 256 KB	More Memory (1 MB) if this doesn't lead to much higher power consumption. Esp. much Flash Memory >= 1 MB for data collections and/or Java Classes		more memory, but obviously this isn't supported in AVR Atmega line yet	good as it is
good	ok		it's perfect the way it is.	?
good	ok		again, perfect.	fine for me
good	Size is ok. Radios need controlled power supply.		possibly as option, but it is fine as is no, it's very good as is. Although we found it very useful to have a third adjustable power supply to generate 5v, 12v, 3.3v, etc. for power hungry peripherals such as GPS receiver	lifetime of a week (with bluetooth power-down) would be nice (do not know how long it is now)
standard battery/charging unit would be great	That would be great. Following the USB adapter idea, power can taken from the USB while developing			?
FAST TIME TO MARKET!!!! i needed it yesterday-- actually the day before yesterday.			I think I covered everything. :-)	-protection against wrong polarization - on/off switch - print name/number of connectors on the pcb