The Evolution of Jini™ Technology In Telematics

By
Judith Arato
and
Steven Knudsen

October 2001

www.psinaptic.com

PsiNaptic Inc.
216, 200 Rivercrest Drive SE
Calgary, Alberta
Canada T2C 2X5
T: 403 720-2531
F: 403 720-2537

Abstract:

Jini™ network technology provides a protocol and system architecture whereby devices and services are added to the vehicle in a completely spontaneous and (often) autonomous way. When first launched, Sun Microsystems' Jini™ technology captured the imagination of automakers looking to seamlessly integrate new product and service offerings into automobiles. At Java™ One 2000, Ford demonstrated a Jini™ technology-enabled vehicle. However, at the time Jini™ technology required more than three megabytes of device storage, limiting its use for in-vehicle applications.

PsiNaptic significantly advanced Sun's initial vision for Jini™ technology on all devices by reducing its footprint to less than 100 kilobytes. Exciting opportunities now exist for this micro-version of Jini™ technology in automotive telematics applications. Furthermore, this technology is applicable to many resource-constrained or mobile devices. Telematics systems can interact with these devices as they enter the vicinity of the car, allowing applications, services, and information to be utilized when and where they are needed.

Two scenarios are presented that exemplify the need for Jini™ technology in automotive systems. In the first, Jini™ technology is applied to the late-binding problem inherent in in-vehicle subsystems. The second scenario addresses the management and control of mobile devices (such as cell-phones and PDAs) that come into contact and interact with automobile systems.

Timelines and motivations for use of this technology platform are discussed.

What is Jini™ technology?

Jini™ network technology provides simple mechanisms that enable devices to form impromptu network communities (Jini™ federations) - communities formed without planning, installation, or human intervention. Devices in a federation may provide services that other member devices may use. Moreover, services and information can be shared between members seamlessly, securely, and reliably.

Jini™ technology uses a lookup service with which devices register their services. A service is implemented by a self-contained Java™ object. A device entering the network first locates the lookup service (discovery phase) and then uploads one or more Java™ objects that implement its services' interfaces (join phase).

Other devices in the federation, whether controlled by a user or acting autonomously, are able to access services via a lookup service. Within the federation, there may be multiple lookup services hosted by one or more devices. Via the lookup service a client device acquires a desired service object and uses that object locally to interact with the device offering the service. The lookup service acts as an intermediary to connect a client looking for a service with that service. A service, because it is a self-contained object, can be offered and utilized independently of its originator, though typically service objects will communicate back to their source for data or for support processing. Once a client is finished with a service, it may discard the service object. This keeps memory requirements low, reduces system management and ensures that services do not become outdated in the hands of the end-user.

Jini™ technology not only defines a set of protocols for discovery, join and lookup, but also a leasing and transaction mechanism to provide resilience in a dynamic network environment. All these mechanisms are combined in Jini™ network technology and can be used to give devices and systems the ability to form powerful spontaneous distributed networks that allow the seamless sharing of information and services when and where needed.

The Java™ programming language is the key to making Jini™ technology work. By using the Java™ programming language, a Jini™ network architecture is secure. The discovery and join protocols, as well as the lookup service, depend on the ability to move Java™ objects, including their code, between Java™ virtual machines.

What advantages does Jini™ offer automotive applications?

Jini™ technology provides a protocol and system architecture whereby devices and services can be added to the vehicle in a completely spontaneous and (often) autonomous way. When a new device or a new service comes into contact with the in-vehicle network, a Jini™ technology-enabled subsystem can discover it and incorporate or use it. In effect, the subsystem can make queries such as: “Do you have a voice interface? Do you have a graphical interface? Do you have an icon?” and so on, and then decide how best to use the new device or service. The subsystem need not necessarily know how the new device or service behaves internally or how it communicates with other in-vehicle and external systems; in true object-oriented fashion, such details are left to the service or device itself.

This eliminates the need to know *a priori* – the communication protocols and other details of operation for devices and services that are introduced late in the design and manufacturing process or thereafter.

What problems can Jini™ technology solve?

a) Late Binding: The introduction of devices and capabilities after the design of a vehicle has been finalized is referred to as "late binding". This is a common occurrence in the automobile industry as automobiles are powered and controlled by multiple subsystems, often proprietary, coupled by a complex network. With the increasing use of electronic components in vehicles, automakers continuously strive to contain costs, whilst trying to incorporate the latest advancements in technology. There is also a need to reconcile the vehicle development cycle with the market drivers of consumer electronics. Adding devices from multiple vendors and services to interact seamlessly with the car telematics systems throughout the design, manufacture, sales and ownership phases is becoming increasingly necessary to future-proof vehicles. Adding a Jini™ technology to the different vehicle subsystems allows for the introduction of devices and capabilities after the design of a vehicle has been finalized, not only addressing the issue of "late binding", but also providing several advantages to upgrading and future-proofing automobiles.

b) Synchronization of Mobile Devices: Solving the "late binding" problem is the first step towards applying a Jini™ technology platform to applications that go beyond the closed environment of in-vehicle systems. As Bluetooth™ becomes more prevalent in mobile phones/ PDAs, telematics control units, and other devices, Personal Area Networks (PANs) or Vehicle Area Networks (VANs) will become common. There will be a need to facilitate and control the seamless interaction between the devices in VANs. Jini™ technology embedded into these resource-constrained devices will enable them to participate in VANs or PANs and autonomously share information and services. Seamless interaction between devices from different manufacturers and the user's ability to easily, quickly, and reliably access device services across physical and logical boundaries will significantly impact the adoption of these devices and future technologies. Future applications include location based and electronic / mobile services linking cars to cars, cars to the freeway infrastructure, or cars to services at businesses, such as gas stations.

Why isn't Jini™ technology being used today?

Although Sun Microsystems Jini™ technology still promises to enable next generation networks, its current size is too large for devices with memory and power constraints. Based on Sun’s Jini™ Technology, PsiNaptic has developed, tested and qualified its *Empromptu* software with a footprint of approximately 100 kilobytes that enables resource-constrained automobile subsystems and mobile devices such as TCUs and cell phones or PDAs to participate as full members of Jini™ federations.

How does *Empromptu* make Jini™ technology feasible in Telematics?

There is an increasing acceptance of both Java™ technology and Bluetooth™ in the auto industry. Java™ technology provides a standard software platform that has been adopted for use in embedded automotive systems. Today’s in-vehicle electronics systems are complex, being comprised of multiple subsystems each with one or more microprocessors. Using Java™ provides a consistent, standard and portable base upon which to coordinate the activities of these systems. Bluetooth™ provides means by which the automobile telematics systems can communicate with mobile devices introduced into the vehicle and over short ranges with external devices.

A small footprint Jini™ technology protocol for resource-constrained devices provides a simple mechanism that enable devices to form an impromptu federation without *a priori* planning, installation or human intervention. Multiple auto subsystems or mobile devices that are Jini™ technology-enabled can announce their presence to the network federation and their information and services immediately become available to other member devices.

PsiNaptic’s Jini™ Technology Implementation: - *Empromptu*

Recognizing the need to provide objects with the ability to form an unlimited network, interacting dynamically and seamlessly, on behalf of people, PsiNaptic developed *Empromptu* software, a small footprint Java™ and Jini™ technology-based platform to improve the delivery of services and enable new innovative applications.

How does *Empromptu* solve the “late binding” problem?

As technology advances, new cars continually feature new systems and innovations. Change and innovation in the auto industry takes time to implement—new systems and technologies can come from multiple vendors and must be integrated in new automotive designs, a costly and time-consuming process. A new component system (such as a new braking system) in a fully developed prototype can take as long as four years to incorporate into a new model. Part of this time is needed to design, build and install production tools to make the new model. There is also time involved with testing the new system on test beds and in pre-production vehicles to see what happens to overall performance.

Automobiles are controlled by multiple subsystems that must coordinate their functions and respond to changing conditions in and outside the automobile. The major systems in an automobile include the power plant, power train, braking system, running gear and others, which are linked to an on-board computer (OBC). Each of these major categories has a number of subsystems, which are connected by a network, typically a controller-area-network (CAN). There are three classes of CANs for automotive electronics in vehicles. The Class C CAN implementation is based on a 500 kbps CAN bus and the SAE J2284 physical and data-link specification. The Class C CAN is suitable for dynamic vehicle systems such as antilock braking and engine control. Class A and Class B CAN implementations support slower data rates (10-125 kbps) and are used for convenience features such as power seats, climate control, and body electronics and diagnostic systems. The OBC, which acts as a hub to these various systems, is capable of translating low-level signals from the different CAN buses.

PsiNaptic can provide in-vehicle solutions for the on-board computer (OBC) interactions with the various car subsystems. Our low cost, small footprint ***Emprompt*u** addresses the co-existence and compatibility problems of in-vehicle subsystems — specifically the synchronization of late changes made to subsystems referred by the automotive industry as the "late binding process". With ***Emprompt*u** on the OBC, the focus is on the vehicle's internal workings, not consumer-based telematics applications, which focus on the driver or passenger.

An OBC interacts with the various subsystems in an automobile, connected by CAN buses. This information is usually make and model specific and often in a proprietary format. With ***Emprompt*u**, the OBC and subsystems in the automobile can interact via the mechanisms specified by the Jini™ specification. The operational and data details specific to each subsystem can be encapsulated in service objects and, thus, hidden from other subsystems. This approach — classic in object-oriented design and implementation — allows systems to change and evolve independent of changes to underlying OBC or subsystem hardware, software, and protocols.

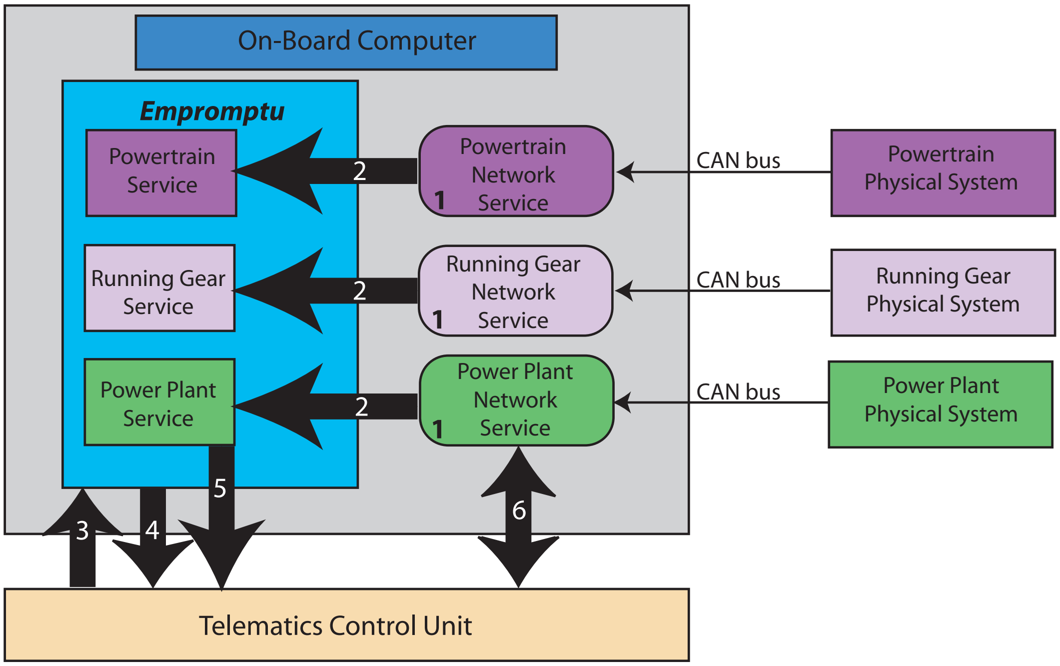

Figure 1: A model for Telematics Control Unit interaction with automotive subsystems using ***Empromptu***.

Figure 1 is intended to be a representative model of how PsiNaptic's ***Empromptu*** could facilitate the interaction between a TCU and an automotive subsystem. An actual implementation may differ from what is being proposed above. The steps indicated in

Figure 1 are interpreted as follows:

1. The Power Train, Running Gear and Power Plant are physical systems. There is a corresponding Network Service for each of those devices. The Network Service is a software representation of a physical device and provides the interface for interacting with that device. For example, the Power Plant Network Service will acquire and translate CAN bus data, making it available in a standard format for other software entities. This will free them from having to be aware of the Power Plant details such as make, model, revision levels, etc. The Network Service is co-located with the ***Empromptu*** Lookup Service (LUS) and both reside on the OBC.
2. Each Network Service joins the LUS by discovering and registering itself with the LUS.
3. The ***Empromptu-***enabled TCU discovers the LUS on the OBC using standard Jini™ technology discovery protocols.
4. The TCU obtains the ServiceRegistrar[1] object and uses its methods to determine if a Power Plant Network Service is registered.
5. The TCU obtains (downloads) the Power Plant Network Service object from the OBC LUS
6. The TCU uses the methods of the Power Plant Network Service object locally (i.e., the object is executed on the TCU) to interact with the physical device.

Benefits

- Flexibility and ease of upgrading to next generation subsystems — addition, removal of components or new code without replacing entire systems.
- Future proofing of major systems, decoupling innovation and product development cycles.
- Cost savings to auto manufacturers — reduced risk of co-existence, compatibility, connectivity issues when reconfiguring hardware or software.
- Subsystem information translated into standard protocol.
- Accessible by auto manufactures, dealerships, service centers.
- Better information of what is happening under the hood.
- Improved customer satisfaction.

How does *Empromptu* facilitate the synchronization of mobile devices in Telematics?

Once a vehicle's subsystems are equipped with ***Empromptu***, the information and services provided by the subsystems can be easily extended to mobile devices within the vehicle. ***Empromptu*** will allow TCUs and mobile devices to share applications and services in a car. The key to introducing new technology in an automobile or telematics industry is commercial potential. The integration of in-vehicle systems with consumer electronics will create commercial potential, as consumers will finally get seamless integration of their devices and services. The initial market drivers are safety related services such as an in-vehicle / remote diagnostics, triggering a 911 call after an accident and measuring / managing a driver's cognitive load.

Many automobile industry leaders and innovators anticipate that the TCU will be an integral part of dynamic Personal Area Networks (PAN). In addition to providing services (traffic, infotainment, etc), the TCU will manage and control other mobile devices in a car. One of the main concerns today is driver distraction. For example, if a cell-phone rings whilst the driver is traveling at a high speed, the TCU will answer the phone advising the caller that the driver is not able to answer the phone and to leave a message. When the car stops, voice messages are displayed on the TCU. This would all be enabled by ***Empromptu***'s ability to allow devices to discover one another and share information and services.

Mobile workforce applications are also market drivers. Business travelers can personalize a TCU in a rental car using their "preferences" service on their ***Empromptu*** enabled PDA (or cell-phone). By simply bringing the two devices into proximity of each other, the TCU could be customized to the user, providing useful information (e-mail, hotel, banking, etc.),

---

[1] The ServiceRegistrar interface is defined in the Jini specification.

specific to the driver. Similarly, a PDA could offer its calendar/appointment service to the TCU, which could then take this information and offer the driver the appropriate maps and driving instructions (this could be particularly useful in the rental car industry).

With ***Empromptu*** enabling the integration of mobile devices with the in-vehicle network, it is possible for the car to always have access to the latest technology, thus offering a kind of "future-proofing" for telematics control unit manufacturers as well as synchronization of telematics services.

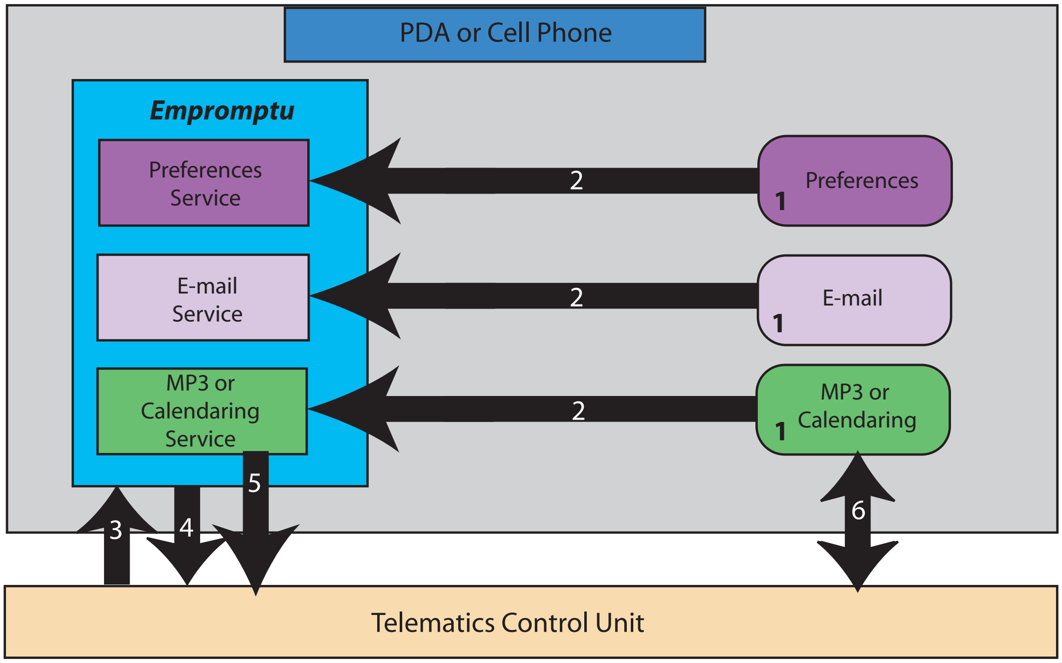

Figure 2: A model for Telematics Control Unit interaction with mobile devices using ***Empromptu***.

Figure 2 is intended to be a representative model of how PsiNaptic's ***Empromptu*** could facilitate the interaction between a cell phone or PDA and an automotive telematics unit. An actual implementation may differ from what is being proposed above. The steps indicated in Figure 2 are interpreted as follows:

1. PDA and/or cell-phones are physical devices that provide services, such as content preferences, e-mail and calendaring.
2. Each application joins the ***Empromptu*** LUS by discovering and registering its service(s) with the LUS.
3. The ***Empromptu*** -enabled TCU discovers the LUS on the PDA using standard Jini™ discovery protocols.
4. The TCU obtains the ServiceRegistrar object and uses its methods to determine if a Calendaring Service is registered.
5. The TCU obtains (downloads) the Calendaring Service object from the PDA LUS.

6. The TCU uses the methods of Calendaring Service object locally (i.e., the object is executed on the TCU) to provide information to the on-board navigation subsystems.
7. The TCU can combine GPS information to a Calendaring entry to provide the driver with directions/map information.

Benefits

- TCUs — useful to any mobile device.
- Improved Safety — meeting customer demands and government legislation.
- Cost savings — product development, enhancements, recall notices.
- Additional revenue — extra value-added services.
- Customer flexibility — any personal device will work in any vehicle without the customer or manufacturer providing any system configuration.

What will *Empromptu* do beyond "late binding" and synchronization applications?

As ***Empromptu*** evolves from subsystems to synchronization with consumer devices in the vehicle, the next step is to extend the information and services as appropriate to other devices that the vehicle may encounter external to the vehicle such as service stations, parking facilities, road systems and other vehicles. These applications will be explored in more detail in another white paper.

Conclusion

Sun Microsystems Jini™ technology has strong applications in the Telematics market; however, its current size is not practical for devices with memory and power constraints. As the many players in the Telematics industry establish the parameters and standards for the telematics industry, ***Empromptu*** can be applied to address the co-existence and compatibility problems of in-vehicle subsystems, providing immediate cost benefits and future proofing technology in automobiles. Cost reductions realized by solving the "late binding" problem should be sufficient enough to justify the increased deployment of TCUs in vehicles paving the way to additional revenue opportunities from extended Telematic services.

Increased presence of sophisticated TCUs and the diversity of mobile devices interacting with automobiles, presents interesting opportunities for automobile and consumer electronics manufacturers to offer new value services. Transparent interaction between devices from different manufacturers and the user's ability to easily, quickly and reliably access device services across network and physical boundaries will significantly impact the adoption of these devices and technologies.

Initial commercial potentials exist for safety related applications such as wireless connectivity after an accident, in-vehicle or remote diagnostics, and enabling a TCU to measure the driver's cognitive load to control other mobile devices in the car. By enabling the integration of mobile devices, it is possible for the car to always have access to the latest technology, thus future-proofing telematics control units.

Longer term, a compelling opportunity exists for business users and auto manufacturers in the realm of brand differentiation and new revenue opportunities by integrating third party, service/content providers (location based, electronic and mobile commerce). This will include linking cars to cars, cars to the freeway infrastructure, or cars to services at businesses, such as gas stations.

The evolution of Jini™ Technology in Telematics
can start today with ***Empromptu*** software
from PsiNaptic.

About The Authors

Judith Arato, Director, Market Development
Tel: (403) 720-2531 x 222
e-mail jarato@psinaptic.com

Steven Knudsen, CTO
Tel: (403) 720-2531 x 227
e-mail sknudsen@psinaptic.com

PsiNaptic Inc.
216, 1200 Riverside Drive S.E.
Calgary, AB T2C 2X5 Canada

PsiNaptic Inc.

PsiNaptic's vision is a world with pervasive, unlimited networks of intelligent objects interacting dynamically and seamlessly, on behalf of people, to improve the delivery of existing services and enable new innovative applications.

PsiNaptic's mission is to be the dominant provider of novel communication technologies that enable real-world pervasive computing applications.

Terminology

Automotive Telematics

Telematics refers to the combination of in-vehicle computing with telecommunications capabilities. Users are able to access and act upon content, applications and services residing on or available via the in-vehicle and external networks. The industry vision is to enhance driver and passenger safety, productivity and security through communication, information and convenience services. This is enabled by the convergence of wireless systems, global positioning and on-board automotive electronics.

Pervasive Computing:

A Pervasive Computing environment is defined by NIST (National Institute for Standards and Technology) as being composed of numerous, casually accessible, often invisible computing devices that are frequently mobile or embedded in the environment and connected to an increasingly ubiquitous network infrastructure which is composed of a wired core and wireless edges.

Embedded Devices

A specialized computer, often hidden from the end user, used in devices and systems found in automobiles, home and office environments, appliances and hand-held units of all kinds, as well as in machines as sophisticated as space vehicles. Operating system and application functions are often combined in the same program. An embedded system implies a fixed set of functions programmed into a non-volatile memory (ROM, flash memory, etc.) in contrast to a general-purpose computing machine.

Java™ Technology

A set of technologies that enable the creation and safe running of software programs in both stand-alone and networked environments. Java™ Technology consists of Java™ language for writing programs a set of APIs, class libraries, and other programs used in developing, compiling and error-checking programs; and a Java™ virtual machine, which loads and executes the class files.

Jini™ Technology

A set of Java™ APIs that may be incorporated an optional package for any Java™ 2 Platform Edition. Jini™ APIs enable transparent networking of devices and services and eliminates the need for system or network administration intervention by a user. Jini™ technology is currently an optional package available on all Java™ platform editions.

Bluetooth™

Bluetooth™ wireless communications technology is a *de facto* industry standard. It is a specification for small-form factor, low-cost, short-range radio links between mobile PCs, mobile phones and other portable devices. Where cables now connect many devices, a wireless Bluetooth™ connection will provide low-cost wireless communications and networking between devices. This enables wireless connectivity to networks and other devices anytime, anywhere. Bluetooth™ is based on a global radio-frequency (RF) standard, which operates on the 2.4 GHz ISM band, providing license-free operation in most of the world.

Bluetooth™ Special Interest Group is an industry group comprising leaders in the telecommunications and computing industries that are driving development and promotion of Bluetooth™ wireless technology and bringing it to market in a broad range of products.

The evolution and promotion of Bluetooth™ wireless technology within Bluetooth™ Special Interest Group (SIG) and in the user community is led by nine promoter group companies: 3Com, Ericsson, Intel, IBM, Lucent, Microsoft, Motorola, Nokia and Toshiba.

References

[1] www.Java.sun.com
[2] www.Jini.org
[3] K. Arnold, et al, The Jini™ Specification,
2nd ed., Reading, Mass.: Addison-Wesley, 2001.

Legal Notice

The information from or through this publication is provided "as-is," "as available," and all warranties, express or implied, are disclaimed (including but not limited to the disclaimer of any implied warranties of merchantability and fitness for a particular purpose). No representations, warranties or guarantees whatsoever are made as to the accuracy, adequacy, reliability, timeliness, completeness, suitability or applicability of the information to a particular situation. The information may contain errors, problems or other limitations. In no event shall PsiNaptic Inc. be liable for any direct, indirect, special, incidental, or consequential damages (including damages for loss of business, loss of profits, litigation, or the like), whether based on breach of contract, breach of warranty, tort (including negligence), product liability or otherwise, even if advised of the possibility of such damages. The entire risk arising out of the use of this information remains with recipient.

Any redistribution or reproduction of any materials or information contained herein is strictly prohibited. This publication and the information may be used solely for personal, informational, non-commercial purposes, and may not be modified or altered in any way. Recipient may not remove any copyright or other proprietary notices contained in the documents and information. Throughout this paper trademarked names are used. Rather than put a trademark symbol in every occurrence of a trademarked name, we state that we are using the names only in an editorial fashion and to the benefit of the trademark owner with no intention of infringement of the trademark.

PsiNode is a trademark of PsiNaptic Inc. Other company names, brand names or product names mentioned herein may be trademarks and/or registered trademarks of their respective owners and companies.