A Proposal for Application-to-Application Network Addressing in Clean-Slate Architectures

Alisson O. Chaves(✉), Pedro Frosi Rosa, and Flávio O. Silva

Faculty of Computing (FACOM), Federal University of Uberlândia (UFU), Uberlândia 38400-902, Brazil
{alisson,pfrosi,flavio}@ufu.br

Abstract. The addressing model presented in current networks has been around for 40 years, has become a standard, and follows the OSI reference model. In this sense, we have a different type of addressing at each layer. The addressing model is part of some research on new internet architectures of the future. These models allow the unified use of different addresses used, reducing computer networks' complexities and improving support for new applications' communication needs on these networks. This work aims to contribute to a simplification of the addressing process in application-to-application communication. With an experimental evaluation, we showcase how our work can benefit clean-slate network architectures.

Keywords: Future Internet · Addressing · Network computing

1 Introduction

The beginning of the concept of addressing in computer networks occurred through the human needs to store information in its memory. In the 70's we already had client computers and a few servers offering services to a company's network. The task of decorating the IP address of these servers was somewhat trivial.

As time passed, the global computerization process systematized across computing solutions that required creating more of these network servers. Thus, the task of memorizing IP addresses started to become more complex, requiring a simpler solution.

Proposals for developing new future internet architectures have been published, demonstrating a good definition of the control plan and data plan's roles. In which devices it influences together with these proposals, the addressing model has been redesigned to support new applications and reduce addressing complexity.

The use of headers in the application layer, among others, increases a large amount of non-useful data in network traffic, increases the use of computational resources for unpacking the package, and making the decision to forward the operating system socket.

L. Barolli et al. (Eds.): AINA 2021, LNNS 226, pp. 783–793, 2021.
https://doi.org/10.1007/978-3-030-75075-6_64

This work assumes that it is possible to further reduce the addressing complexity by creating a specific Virtual Network Interface Card (vNIC) for each communication channel between clients and server applications - a service known as a socket. The practical applications of this solution are reflected even in the form of data sent by the applications.

This work contributes to supporting the development of agnostic applications to the network, reducing network traffic consumption, simplification in addressing applications, and an increase in the number of hosts available on the internet. The solution natively provides interoperability between different architectures and applications developed for possible heterogeneous environments. The remainder of this paper is organized as the following. Section 2 presents a brief history of the use of names in the addressing representation process and the concept of ETArch architecture. Section 3 discusses related works. Section 4 describes the application-to-application addressing proposal for this work. Section 5 presents the details of our experimental evaluation. Section 6 analyzes the results obtained with the tests. Finally, Sect. 7 presents concluding remarks.

2 Background

In this section, we will talk about the concepts and architectures used as principles for developing this project.

2.1 Domain Name Service

Dated 1982, a solution was introduced using the file *hosts.txt*, a text file that contains a list correlating an IP address to any name. This simple solution was implemented on the internet for all connected devices, and synchronization was done by transferring the file hosted on the SRI Network Information Center (SRI-NIC). The solution could also be used on local networks [9].

The concept of addressing is widely used in information technology, from memory locations, process queues to store information on a disk. After the solution was defined, in addition to the continuous growth in the number of services provided by the network, there was a great growth in the types of objects to be addressed in a network, such as switches and routers, it would literally be possible to have at least one name for each one of the 4,294,967,296 IP addresses available worldwide.

Managing, maintaining, organizing, and distributing this list has now become a complex and costly process for networks. The transfer of this file could cause degradation in the internet backbones and the large workload applied to the SRI-NIC server, and the number of changes to the *hosts.txt* file [9].

Works published in the late 1970s and early 1980s originated RFC 882 and 883, which define the concepts, specifications, and model for implementing domain names, which do not replace the use of the *hosts.txt* file, but rather define a global pattern of use [9].

From these definitions today, the DNS has undergone several improvements. However, for this work, we will describe some features.

2.1.1 Language

The language adopted for digital representations of information is ASCII, restricted to characters globally used in the vast majority of languages and some characters' reserves for special functions. There is no size limitation on domain names.

2.1.2 Hierarchy

In the hierarchical and tree operating mode, which in addition to being able to segment management into branches, also allows us to consult directly with branches, which reduces the need to consult the entire list of name-domain relationships, this increases the name resolution performance and decreases the load on domain servers and network backbone.

The branches are organized using the dot character [.]. For example: *mehar.facom.ufu.br*, where *mehar* is a branch of *facom* that is part of the *ufu* branch that participates in the trunk *br*.

2.1.3 Uniqueness

The principle of exclusivity was necessary for the DNS service to guarantee a single, impartial and secure access. Root, trunks, branches, sub-branches, and all their entries must be unique and exclusive throughout the internet. For the registration process of these names to be organized, the IANA - Internet Assigned Numbers Authority was created, which, among several functions, performs the registration of IP addresses on the internet and manages the assignment of domain name root servers, this way, professional, fair and neutral [5].

2.2 ETArch Architecture

The ETArch architecture project uses the concept of assigning titles between the components present in communication. The following are defined: the entities that are participants in the communication, the title, which is the unique form of identification independent of the topology, and the workspace, which is the means of interconnection for communication [3, 13].

Technically, the implementation of ETArch would be based on addressing the entity and workspace. By addressing fields available in the link layer of the OSI reference model, the entities and workspace are represented. An abstraction of the network, transport, session, and presentation layers is carried out, simplifying the traffic on the network bringing gains concerning performance and addressing. Figure 1 presents the basis of ETArch architecture and its components.

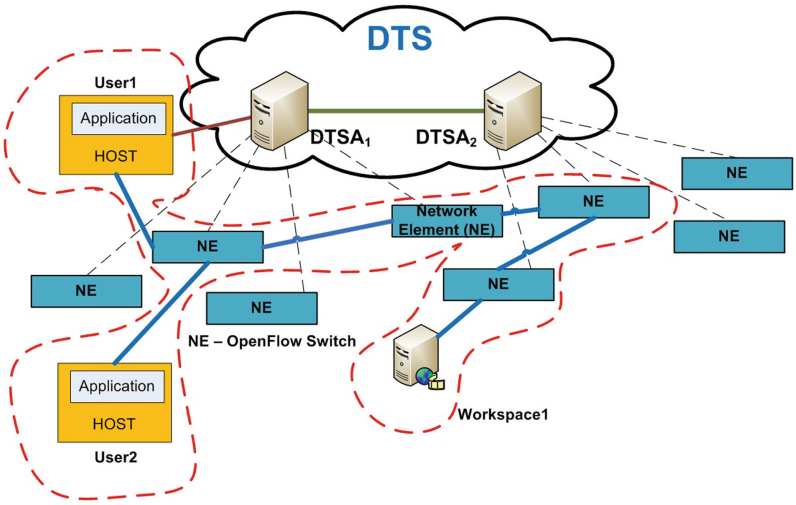

Fig. 1. ETArch architecture [12].

Using the implementation of this project, contributions were made to the definition of an addressing model.

2.3 Addressing

IP addressing solves the problem of finding a device presence on the internet. It represents using decimal numbers the addressing in the network layer of the OSI reference model. Its version 4 allows approximately 4 billion addresses, and since 2014 it has been running out, so after the creation of version 6, we have more than 340 undecillion of possible addresses. However, the use of a new protocol is only possible by updating all devices and operating systems present on the internet, making the process slow. Even today, several networks do not have IPv6 addresses.

With a size of 65536 possibilities, the port is known to also represent, through decimal numbers, the context of an application within a device. It is considered an application layer address, but the transport layer also uses it. They form the socket with the IP address, which is responsible for allowing the communication of different applications between one or several client/server devices simultaneously.

The MAC address, which has more than 281 trillion addresses, represents the address in the link layer of the OSI model through hexadecimal numbers and, by definition, has its use controlled for registered manufactures of network devices, where a unique MAC address is installed to each Network Interface Card manufactured in the world.

3 Related Work

There are different research related to the Future Internet. Several works have proposed a model of addressing networks.

The work from Seoul [2], proposes a division in the semantics of network locators and identifiers, which today is part of the addressing of the network layer, bringing improvements in support of mobility multi-homing.

Geoff Houston [4] presents the properties uniqueness, consistency, persistence, trust, and robustness, which are essential for the network operation. He also describes that the characteristic of the combination of functions present in the IP address of *who*, *where*, and *how* not only makes the use of the network more efficient, they are also the cause of the complexity existing on the internet today. Problems such as mobility, the addressing system's granularity, and the lack of identification of complete paths through the network are research challenges.

The NDN Project [14] proposes using identifiers hierarchically structured in all communication elements of the network. Instead, the names assume the role of identification and can reference the data more simply. In one of its principles, end-to-end, it allows the development of more robust applications in network failures. Some characteristics are unique and hierarchical names of variable length and addressing by names.

Researchers at MIT [10] present the principle of the end-to-end argument that suggests that the functions applied to the network can be redundant or costly, since in many services available on the internet today, they perform these functions directly at the application layer. Functions like Delivery Guarantees, Secure Transmission of Data, Duplicate Message Suppression, and point identification are examples of possible unnecessary use of resources that we use today.

Proposals for the development of new *clean-slate* architectures such as Mobiliy-First [11], NEBULA [1], and ETArch [12], have been published, demonstrating a good definition of the roles of the control plan and data plan and which devices it influences.

However, when we look at everyone involved in the communication, these proposals are focused only on network devices, not considering the operating systems of the final devices, thus requiring applications that consume network services to know the structures of the network and also to have greater interaction with the OS at the level of identifications in the application layer.

4 Proposal

Future Internet projects reduced the number of headers to only the one present in the link layer. Because it does not interfere with the application layer protocols, the headers must still exist for traffic to be redirected to the device's correct application.

Aiming to make usability in new internet technologies of the future simpler, in this work, we contemplate an effective communication between the new network architectures and the operating systems to provide transparency in the application data traffic.

Figure 2 presents an overview of the component of our solution. In this proposal, we separate the components into layers with their respective functions. In the application component layer, we have several applications developed for different types of architectures. Identification of these applications' communication is sent to the converter component layer, where it will be converted into a MAC address.

This MAC address is now used to create a Virtual Network Interface Card of the *macvlan* type in bridge mode for each application communication. Now the data is sent on the physical interface available on the host using its own identification on the network.

Without hurdling the functionalities described in this paper about the DNS, it is proposed to create a virtualized NIC for each application socket within the device's operating system. Thus, when an application invokes a socket to use through an application, this vNIC is created and used solely and exclusively for that application.

As sockets' concept was abstracted, the traffic of an application through this vNIC can be in raw mode, that is, send only the useful data related to the context of the application.

For example, after accessing the application socket and creating the vNIC, a chat application can traffic only the bits referring to the message's text. The management and use of the operating system's network sockets would be simpler, and the volume of data for the same application compared to today's networks would be less, which provides faster access to the services available in these architectures.

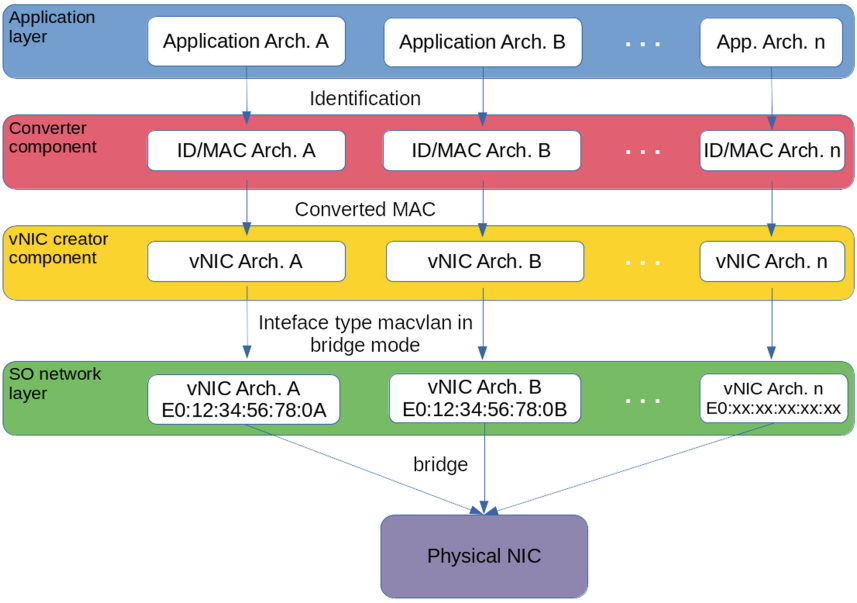

Fig. 2. Solution components.

In an application, a message exchange module and a video conference module belonging to the same server must have unique and hierarchical identifiers as a name, for example: *msg.chat.mehar.facom.ufu.br* and *vid.chat.mehar.facom.ufu.br*.

5 Experimental Evaluation

This section will demonstrate that the solution proposed in this work can work all the clean-slate network architecture cited in Sect. 2. To do the experimental evaluation of our proposal, we selected the ETArch network architecture that our research group proposed.

The addressing fields available in the ETArch architecture are the Destination MAC Address and Source MAC Address fields, which have a size of 48 bits. Of these 48 bits, the 7 bit is reserved to identify whether the address is globally unique or locally administered, and bit 8 is used to check whether the address is unicast or multicast.

Using the Linux OpenSuSE Leap 15.0 64bit operating system, the use of the network driver virtualization library called *macvlan* was defined. Macvlan was developed to create a communication channel between the physical NIC and virtual machine [7].

In this work, the use of the concept of domain names for use in titles and workspaces with all their language, hierarchy, and exclusivity functions were defined. A configuration file indicates the device's title and on which physical network interface the vNIC will be attacked.

The software created reads this information and uses the MAC address conversion library's domain name to create the vNIC corresponding to the device.

The library reserves the first 8 bits as *1110 0000* which will result in hexadecimal *E0* and converts the domain name to hexadecimal using the default unsecured hash function *CRC32 CRC16* [8] with 40-bit output, thus totaling the 48 bits present in the link-layer addressing fields.

After creating the workspace, the application must make a call informing the name of the workspace to the conversion software that will have "<device_title>+ <workspace>" input and 48-bit address output.

At this point, the software developed creates a vNIC *macvlan* type in bridge mode, and now the application can open a socket and send data in raw mode through this interface.

5.1 Test Environment

The proposal's test environment has all the necessary participants for the complete functioning of an ETArch network. DTSA is hosted in the MEHAR research data center. In another network connected via the internet, 2 x wireless TPLink TL-WR1043ND routers will be used, which had the operating system modified to OpenWRT 18.06, with the installation of the OpenVSwitch package and OpenFlow 1.0 protocol enabled. Connected to each of these wireless routers, a computer runs the Linux OpenSuSE Leap 15.0 64bit operating system where the developed applications will be implemented.

5.2 Experiment Description

Figure 3 shows the proposal for raw data sending on an application-to-application socket. In the host *alisson.ufu.br* we have two applications, and each application has two different services, and for each service, a corresponding vNIC with unique addressing would be created. These services communicate with the client host. We would have

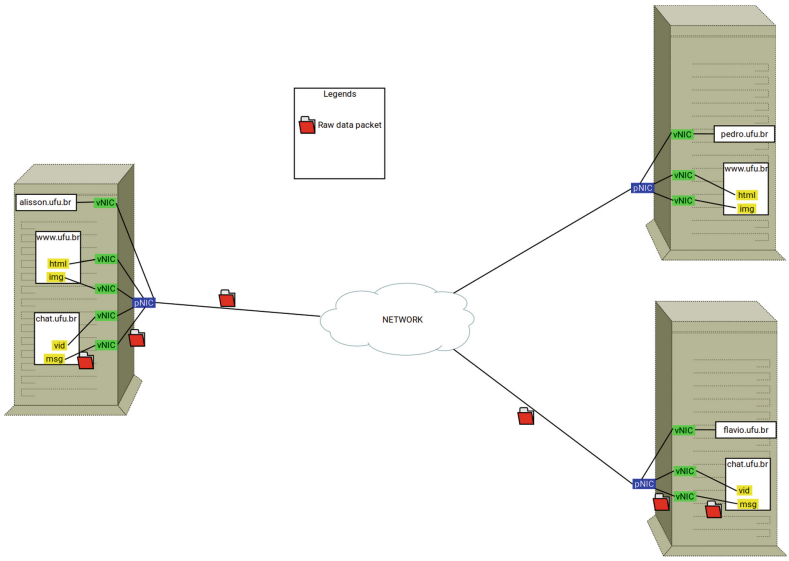

Fig. 3. Raw data sending on application-to-application socket.

access to the respective applications and services provided. Thus, a data package has its destination, the unique identification of the service delivered by the application.

For the tests, a simple ETArch based chat application was used, which sends through the interface intended that application socket in raw socket mode only the pure text to be sent in the message.

To better illustrate the proposal, a 162.2 MByte file was sent through the test structure described above in three network configuration scenarios: 1500 Bytes MTU, 500 Bytes MTU and 130 Bytes MTU. The different MTU sizes simulate sending packets of large and small sizes, and as a means of comparison with a TCP/IP application, we carry out the transmission of the same file through the simple File Transfer Protocol (FTP).

6 Results and Analysis

To perform the tests, we created the vNIC Creator and Converter components. We also used an ETArch chat application that performs the sending of pure data were used. Figure 4 shows the execution of the software developed in two hosts. It checks the configuration file, creates and verifies the interface related to the host identification. It also creates and verifies a workspace and carries out the exchange of messages in the chat application.

Figure 5 and 6 demonstrate the capture of packets sent by the chat application from both hosts, as seen, not being necessary the process of padding the message until completing 46 bytes in the data field of According to the standards of the Ethernet protocol, [6] this allows a reduction in the data sent over the network.

Fig. 4. Terminal configuration, execution and chat test.

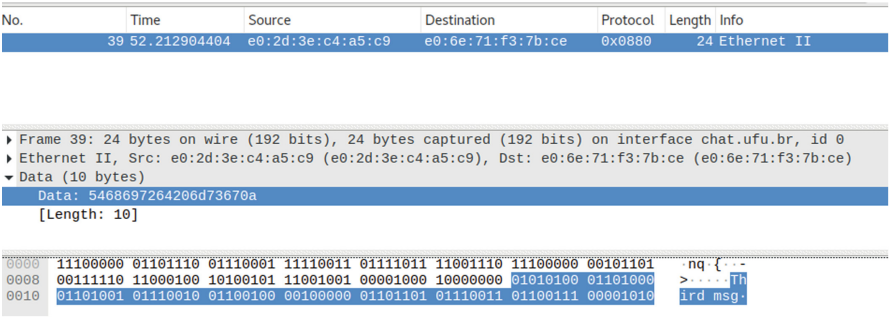

Fig. 5. Send packet capture on first computer (alisson).

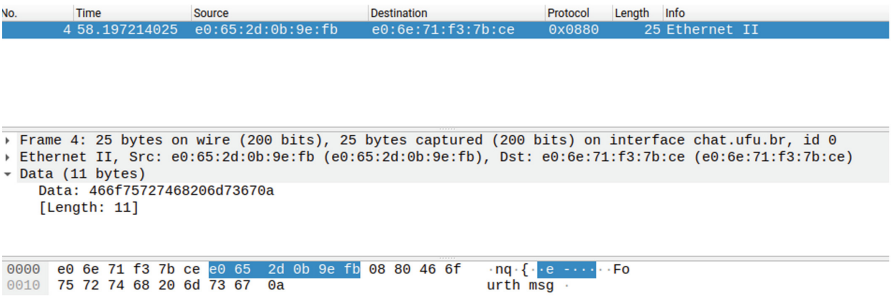

Fig. 6. Send packet capture on second computer (hp).

Figure 7 compares the amount of data sent over the network using the FTP protocol and our proposal concerning the original information contained in the file. The amount of protocol control data plus the size of the PDU layers of the TCP/IP model's layers represent the total send data in the network: 6.40%, 18.08%, and 55.88%, respectively in the MTU sizes of 1500, 500, and 130. In the proposed model, we do not have protocol control data, and the fields used as the combined PDU represent the total send data in the network: 0.92%, 2.72%, and 9.72% respectively in the MTU sizes of 1500, 500, and 130.

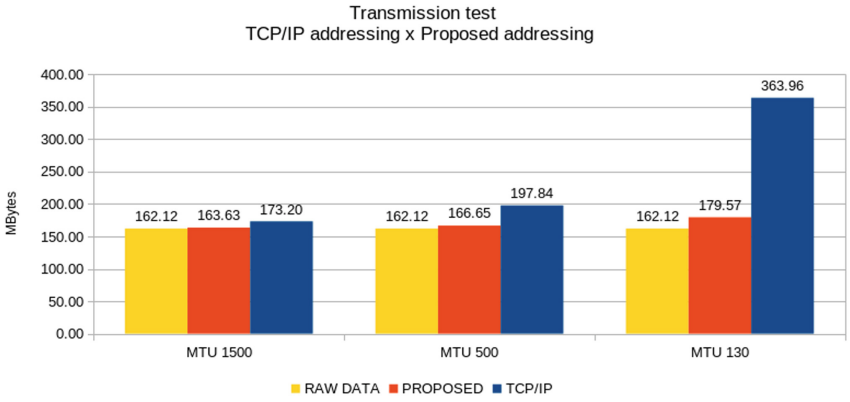

Fig. 7. Transmission test. TCP/IP addessing x Proposed addressing.

In the proposed model, we do not have protocol control data and the fields used as the combined PDU represent the total send data in the network: 0.92%, 2.72% and 9.72% respectively in the MTU sizes of 1500, 500 and 130.

7 Conclusion

Future Internet research projects create a new world of possibilities, which must be studied in depth. The ETArch architecture defines a title model that, among several gains, we can mention the abstraction of layers 3 and 4 of the TCP/IP reference model.

As a contribution, a unique addressing model was defined for the entire TCP/IP stack through the fields currently available in an ethernet protocol gained with the reduction in the minimum size of an ethernet frame.

As an objective, we have also defined a model for developing new applications where it is only necessary to send data through the network interface related to the application socket's name. Using the developed application, we created a channel that allows the creation and maintenance of these interfaces.

A simple sample application of a clean-slate network architecture was used to test this work's assumptions and achieved the expected objectives.

Future work must adapt the interface maintenance system as a module of the Linux operating system. The development of more complex applications in client / server architectures would be a powerful object of study.

Another future objective is to apply this model to other clean-slate network architectures, thus ensuring interoperability between the projects' concepts. Another possibility would be tested on legacy networks and the current internet to verify the applicability of the concepts defined here in the historical load of internet development.

References

1. Anderson, T. et al.: The nebula future internet architecture. In: SPRINGER. The Future Internet Assembly, pp. 16–26 (2013)
2. Choi, J., et al.: Addressing in future internet: problems, issues, and approaches (2008)
3. Corujo, D., et al.: Enabling network mobility by using IEEE 802.21 integrated with the entity title architecture (2013)
4. Huston, G.: Addressing the Future Internet. The ISP Column (2007)
5. IANA - About us. https://www.iana.org/about/. Accessed 12 Jan 2021
6. Kurose, J., Ross, K.: Computer networking: a top-down approach. CERN Document Server (2017). https://cds.cern.ch/record/2252697. Accessed 12 Jan 2021
7. Liu, H.: Introduction to Linux interfaces for virtual networking (2018). https://developers.redhat.com/blog/2018/10/22/introduction-to-linux-interfaces-for-virtual-networking/. Accessed 12 Jan 2021
8. Miller, F.P., et al.: Cyclic redundancy check: computation of CRC, mathematics of CRC, error detection and correction, cyclic code, list of hash functions, parity bit, information ... Cksum, Adler- 32, Fletcher's Checksum. Alpha Press (2009)
9. Mockapetris, P., Dunlap, K.J.: Development of the domain name system. In: Symposium Proceedings on Communications Architectures and Protocols, Association for Computing Machinery, pp. 123–133 (1988). https://doi.org/10.1145/52324.52338
10. Saltzer, J.H., et al.: End-to-end arguments in system design. ACM Trans. Comput. Syst. **2**(4), 277–288 (1984). https://doi.org/10.1145/357401.357402
11. Seskar, I., et al.: Mobilityfirst future internet architecture project. In: ACM. Proceedings of the 7th Asian Internet Engineering Conference, pp. 1–3 (2011)
12. Silva, F.: Endereçamento por título: uma forma de encaminhamento multicast para a próxima geração de redes de computadores. Tese (Doutorado) — Universidade de São Paulo (2013)
13. Silva, F., et al.: Entity title architecture extensions towards advanced quality-oriented mobility control capabilities. In: 2014 IEEE Symposium on Computers and Communications (ISCC), pp. 1–6 (2014). https://doi.org/10.1109/ISCC.2014.6912459
14. Zhang, L., Estrin, D., Burke, J., Jacobson, V., Thornton, J., Smetters, D., Zhang, B., Tsudik, G., Massey, D., Papadopoulos, C., et al.: Named data networking (NDN) project. Technical report NDN-0001, Xerox Palo Alto Research Center-PARC 2010; 157:158 (2010)