

GENESYS: A Cross-Domain Architecture for Embedded Systems



AUTHORS

R. Obermaisser, H. Kopetz,
B. Huber, C. El-Salloum (TUV)
R. Zafalon (ST Microelectronics)
F. Auzanneau (CEA)

K. Kronlöf (Nokia)
P. Millet (Thales)
M. Borth (ESI)
C. Couvreur (IMEC)
N. Suri (TUD)

S. Campos (ESI-Tecnalia)
E. Ovaska (VTT)
M. Gödecke (Infineon)



ARTEMIS

The ARTEMIS approach is to cut barriers between application sectors [...]

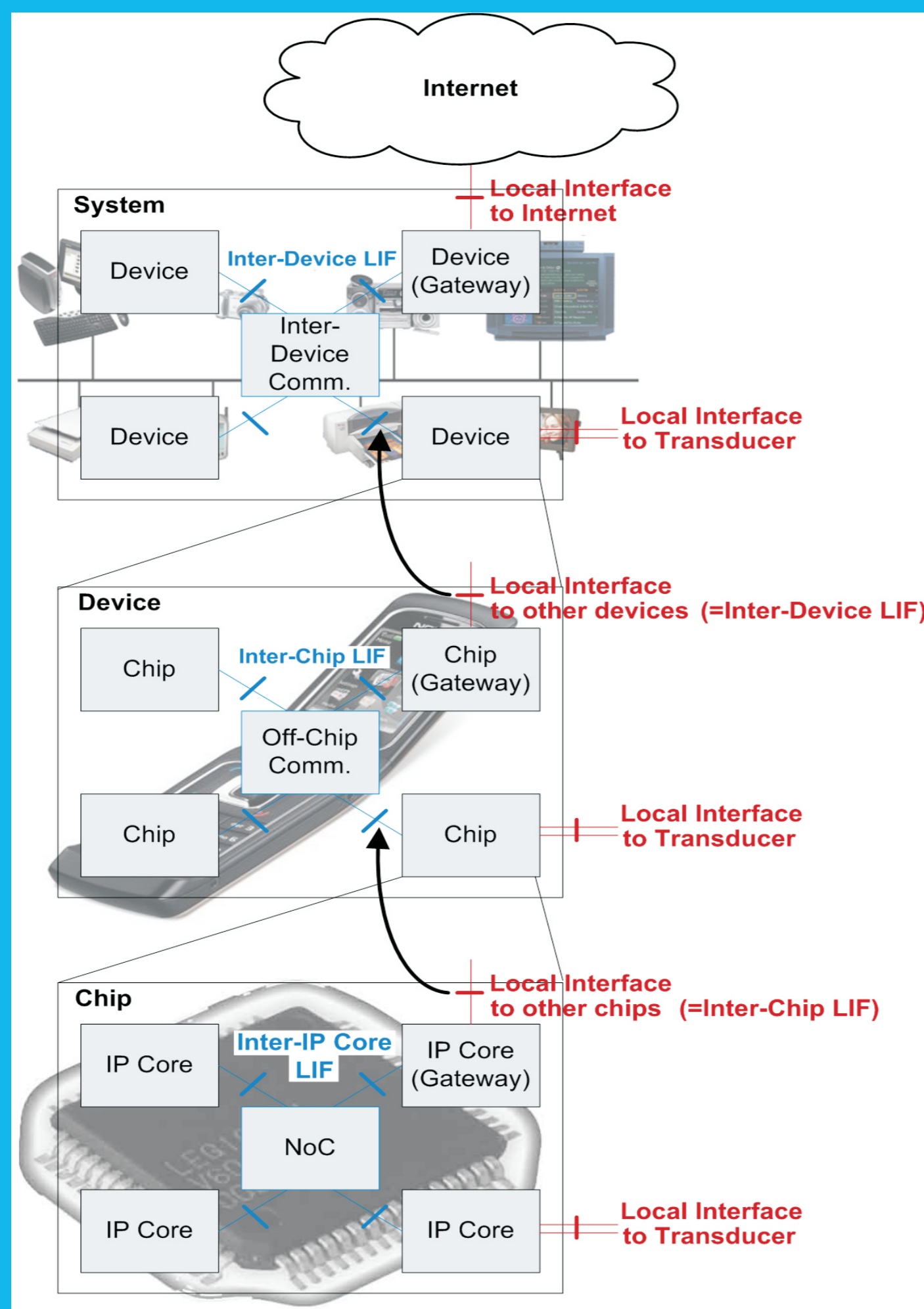
This will be achieved by specifying an ARTEMIS reference architecture that can support product development in a diversity of application domains, for example in automotive, aerospace and nomadic.



[ARTEMIS the European Technology Platform for Embedded Intelligence and Systems, Annual Conference, 2005], p.10

GENESYS

- ✎ Development of an architecture candidate for ARTEMIS
- ✎ Cross-domain architecture for embedded systems which is suitable for multiple application domains (Automotive, Industrial Control, Avionics, Mobile Systems, Consumer Electronics)
- ✎ Devise solutions for technological challenges identified in ARTEMIS including
 - Composability
 - Security
 - Diagnosis
 - Evolvability
 - Networking
 - Robustness
 - Energy efficiency
 - Integrated resource management
- ✎ GENESYS provides a framework for the seamless integration of existing and new services and components across application domains



- ✎ Distinction of three integration levels because of substantially different service characteristics (e.g., communication bandwidth)
 - System (open or closed)
 - Device
 - Chip
- ✎ Component Interface Structure
 - *Linking Interface (LIF)* for the integration of components (technology independent, abstracting from component-internals)
 - *Local Interfaces* to the component environment, such as other subsystems or transducers (can be technology-specific)
- ✎ Interconnection of Integration Levels: Local interface of a gateway becomes LIF of the next integration level

CONSOLIDATED CROSS-DOMAIN ARCHITECTURAL STYLE

- ✎ Rules and guidelines for the partitioning of a system into sub-systems and for the design of interfaces
- ✎ Definition of architectural principles (e.g., ensuring error containment, partitioning of the system along precisely specified interfaces)
- ✎ Avoidance of property mismatches
- ✎ Constrains an architecture in such a way that the resulting system meets the ARTEMIS challenges

CROSS-DOMAIN DEVELOPMENT METHODOLOGY

- ✎ Modeling, evaluation and validation of platform services and embedded systems based on the reference architecture template
- ✎ Measurable quality characteristics

REFERENCE ARCHITECTURE TEMPLATE

- ✎ Description of platform services
- ✎ Generic component libraries
- ✎ Platform service specifications e.g. communication services, diagnostic services, security services, and resource management/reconfiguration services

PROJECT PARTNERS

Vienna University of Technology, Institute of Computer Engineering; STMicroelectronics; Commissariat à l'Énergie Atomique; Nokia; Thalesgroup; Embedded Systems Institute; IMEC; Technische Universität Darmstadt; Fundacion European Software Institute; VTT; Infineon Technologies; Centro Ricerche Fiat; TTTech Computertechnik; Alma Mater Studiorum - University of Bologna; Université Joseph Fourier Grenoble 1; Fraunhofer IGD; Technische Universität München; Vytauto Didziojo Universitetas; Ikerlan S. Coop.; Budapesti Muszaki és Gazdaságtudományi Egyetem; Universidad Politécnica de Madrid; NXP Semiconductors Netherlands; Volvo Technology