Project: ISOLDE: customizable Instruction Sets and Open Leveraged Designs of Embedded riscv processors

Reference number: 101112274

Project duration: 01.05.2023 - 30.04.2026

Work Package: WP6: Open-Source Strategy, Business Models, Exploitation and Dissemination

Deliverable: **D6.5**

Title: Overview of planned IP releases via Virtual Repository and other distribution mechanisms

Type of deliverable: Data

Deadline: 30.04.2025

Creation date: 31.01.2025

Authors: Nicoleta Capbun, BEIA
Angela Ionita, BEIA
George Suciu, BEIA
Mari-Anais Sachian, BEIA
Catalin Ciobanu, IMT
Jaume Abella, BSC
Vicente Nicolau Gallego, FENTISS
Francesco Conti, UNIBO
Sven Mehlhop, OFFIS
Alessio Burrello, POLITO
Daniele Jahier Pagliari, POLITO
Matteo Perotti, ETH

Involved grant recipients: BEIA Consult International (BEIA)
Institutul National de Cercetare-Dezvoltare Pentru Microtehnologie (IMT)
Barcelona Supercomputing Center (BSC)
Fent Innovative Software Solutions S.L. (FENTISS)
University of Bologna (UNIBO)
OFFIS e.V. (OFFIS)
Politecnico di Torino (POLITO)
Eidgenössische Technische Hochschule Zürich (ETH)

Contacts: Nicoleta Capbun, BEIA, nicoleta.capbun@beia.ro

Table of Contents

Executive Summary

I Introduction

The ISOLDE project aims to significantly address the RISC-V growing demand: by the end of our project, we will have high-performance RISC-V processing systems and platforms at least at Technology Readiness Level (TRL) 7 for the vast majority of the proposed architecture components, with the expectation that 2 years after project completion, ISOLDE's high-performance components will be used in industrial quality products. Moreover, the project extends its expertise to the realm of computing by spearheading the development of multi-precision Vector processing units integrated with RISC-V cores. This initiative seeks to enhance computing efficiency for specific applications while furnishing Europe and the open-source community with innovative solutions to computational challenges.

This document describes the virtual repositories of ISOLDE IP as a summarizing work performed within ISOLDE WP6 – Open – source Strategy, Business Models, Exploitation and Dissemination in Task 6.1. A consolidated version of this Deliverable provides information and can demonstrate that networking within the realm of RISC-V, open-source technology, embedded processor development, and SoC application design can greatly expand ISOLDE reach and impact.

I.1 Definitions and Acronyms

Open-source Distribution	The process of providing software that combines a large set of open-source components into a cohesive project or product, which can be freely used, modified, and shared.
Business Model	A strategic plan outlining how a company creates, delivers, and captures value, both economically and socially. It includes customer segmentation, revenue streams, value propositions, and cost structures.
GPL (General Public License)	A widely used free software license that guarantees end users the freedom to run, study, share, and modify the software.
RISC-V	An open-source instruction set architecture (ISA) based on established reduced instruction set computing (RISC) principles.
Vector Processing	A computational approach where a single instruction operates on multiple data points simultaneously, increasing efficiency, particularly in tasks like matrix multiplications and machine learning applications.
RVV (RISC-V Vector extension)	A set of instructions under the RISC-V ISA designed to perform vector operations, enabling more efficient processing by handling large vectors with a single instruction.
Ara	A processor design that works in tandem with the CVA6/Ariane core, designed to efficiently accelerate the computation of long vectors and demonstrate the potential of RVV.
ASIC (Application-Specific Integrated Circuit)	A type of integrated circuit customized for a particular use, rather than intended for general-purpose use.
Digital Sovereignty	The concept that individuals, organizations, and governments should have control over their own digital data and infrastructure, rather than being dependent on foreign entities.

Open-source Hardware	Hardware whose designs are made publicly available so that anyone can study, modify, and distribute the hardware and its design.
PULP Platform	A platform developed by ETH Zürich that encompasses a set of open-source, scalable, and energy-efficient processor cores for high-performance and low-energy computing systems.
CVA6/Ariane, CV32E40P/RI5CY, Ibex/ZeroRiscy	Processor cores designed by ETH Zürich, commonly used in RISC-V based computing systems.
DevOps	DevOps is the integration and automation of the software development and information technology operations.
VHDL	VHDL is a hardware description language that can model the behavior and structure of digital systems at multiple levels of abstraction, ranging from the system level down to that of logic gates, for design entry, documentation, and verification purposes.
OCI	Oracle Cloud Infrastructure
API	Application Programming Interface
AES	Advanced Encryption Standard
FPGA	Field-Programmable Gate Array
GCC	GNU Compiler Collection
ONNX	Open Neural Network Exchange
TF Lite	TensorFlow Lite
ML	Machine Learning
LLVM	LLVM, also called LLVM Core, is a target-independent optimizer and code generator
SIMD	Single Instruction, Multiple Data
MLIR	Multi-Level Intermediate Representation is a unifying software framework for compiler development
ISA	Instruction Set Architecture
IP	Intellectual Property
REST	Representational State Transfer
SBOM	Software Bill of Materials
SPDX	System Package Data Exchange

I.2 General information

ISOLDE stands for “High Performance, Safe, Secure, Open-Source Leveraged RISC-V Domain-Specific Ecosystems.” The project commenced in early 2023 and involves 38 partners from 8 different European countries. It is coordinated by Infineon Technologies AG and has received significant funding from the European Commission under the Horizon Europe program. The current project represents a transformative initiative within the European Union, aiming to accelerate the digital transformation across economic and societal sectors.

At the heart of ISOLDE is the development of high-performance RISC-V processing systems. These systems are designed to be at least at Technology Readiness Level (TRL) 7 for the majority of building blocks, demonstrating their applicability in key European application domains such as automotive, space, and IoT. It could be a testament to the EU's commitment to achieving digital autonomy and fostering a green, climate-neutral future.

A virtual repository was created together with TRISTAN project and a Unified Access Page has been set-up (https://github.com/openhwgroup/tristan-isolde-unified-access-page) ISOLDE partners are updating the common repository by forking and editing the html file, then committing based on their own branch.

I.3 Purpose and scope

This deliverable outlines the architecture, implementation, and operational details of the virtual repository intended to centralize, manage, and distribute artifacts produced during the ISOLDE RISC-V project. It focuses on the GitHub features such as:

- Repository structure and technologies.
- Continuous Integration and Continuous Delivery [CI & CD] – two essential components of both agile and DevOps toolchain that guide to successful software development integration.
- Artifact versioning and traceability.
- Access control and license management.
- Mechanisms for distribution across stakeholders (partners, users).

I.4 Key Components Analysis

Virtual Repository architecture

- Hybrid between GitHub, GitLab [Git-based code repository] and artifact repository.
- Storage Support – source code, compiled binaries, test logs, configuration files documentation.
- Backups and Redundancy – use of cloud-based solutions or mirrored servers for fault tolerance.

Supported Artifact Types

- RISC-V binaries (toolchains, firmware).
- Hardware description files (e.g., Verilog/VHDL).
- Docker images or OCI-compliant containers.
- Test suites, coverage reports, documentation.

Distribution Mechanisms:

- Public/private release channels.
- Secure API access (REST or GraphQL).
- Automated deployment triggers for new versions.
- Use of CI/CD pipelines to validate before release.
- Role-based access for internal/external users.

Version Control and Traceability:

- Semantic versioning scheme (e.g., MAJOR, MINOR, PATCH).

- Hashing and signatures for binary integrity.
- Metadata tagging contributor, date, commit link, test status.
- Cross-referencing artifacts with project milestones and work packages.

I.5 Tools and Technologies

Choices involve:

- Version Control: GitLab / GitHub.
- CI/CD: Jenkins, GitHub Actions, GitLab CI.
- Artifact Repository: JFrog Artifactory, Nexus Repository.
- Container Registry: DockerHub, GitLab Container Registry.
- Security: GPG signing, OAuth2-based access control, SBoM compliance.

Challenges Identified:

- Ensuring interoperability between heterogeneous toolchains (hardware/software).
- Licensing complexity for redistributing third-party components.
- Synchronizing versions across multiple repositories.
- Balancing accessibility with intellectual property protection.

I.6 Recommendations / Next Steps

- Establish periodic audits to ensure repository health and policy compliance.
- Expand user documentation for onboarding and artifact consumption.
- Investigate future support for SBOM and SPDX licensing formats.

- Prepare migration/export strategy for post-project sustainability.

II Foundation infrastructure

This section summarizes public IPs resulting from ISOLDE project work packages and provides the link to their individual repositories.

II.1 WP2 Open-source Foundation Cores

Partner	IP name	Link to repository
GSL	GRLIB IP library	https://www.gaisler.com/getgrlib
TDIS	CVA6 processor extensions	https://github.com/openhwgroup/cva6
ETHZ	Interface between CVA6 and Vector accelerator	https://github.com/pulp-platform/ara/tree/mp/xif

II.2 WP3 Accelerators and Extensions

Partner	IP name	Link to repository
BSC	Safety-related Traffic Injector (SafeTI)	https://github.com/bsc-loca/SafeTI/

Partner	IP name	Link to repository
BSC	Safety-related Statistics Unit (SafeSU)	https://github.com/bsc-loca/SafeSU/
TRT	Context-aware monitoring (CA-PMC)	https://github.com/ThalesGroup/cva6-context-aware-monitoring
UNIBO	Root-of-Trust Unit Design and Interface with RISC-V Host Processor (TitanCFI)	https://github.com/pulp-platform/opentitan
UNIBO	Tensor Processing Unit (TPU)	https://github.com/pulp-platform/pulp_cluster/tree/lg/isolde
ETHZ	Vector Processing Unit (with multi-precision capabilities) (VPU)	https://github.com/pulp-platform/ara
IMT	SIMD/Vector Accelerator	https://github.com/alex2kameboss/MatrixAccelerator
IMT	Scratchpad memory	https://gitlab.com/catalin.ciobanu/PolyMem_System_Verilog
IMT	Number Theoretic Transform Algorithms for Post Quantum Cryptography (NTT)	https://github.com/sirazvan/twister_FFT
IMT	Fast Fourier Transform Algorithms for SIMD and Vector Accelerators (FFT)	https://github.com/sirazvan/twister_FFT
TUI	SIMD/Vector Accelerator	https://github.com/tui-compute/exp

II.3 WP4 System Software, Development Tools and Automation

Partner	IP name	Link to repository
POLITO	MESSY (Multi-layer Extra-functional Simulator in SYstemc)	https://github.com/eml-eda/messy
POLITO	Toolchain for Hardware-aware Neural Network Optimization	https://github.com/eml-eda/match https://github.com/eml-eda/plinio
POLITO	Integrate system level simulators with extra-functional properties	https://github.com/eml-eda/messy
UPV	BAFFI (Bit-accurate FPGA fault injector: dependability assessment and verification of FPGA prototypes)	https://gitlab.com/selene-riscv-platform/DAVOS
IMT	Compiler Support for SIMD/Accelerator	https://github.com/alex2kameboss/MA-riscv-gnu-toolchain
BEIA	Secured RISC-V Processor with Cryptographic Accelerator	https://github.com/ISOLDE-Project/AES-256

Partner	IP name	Link to repository
UPB	Run-Time Environment for end-users	https://gitlab.upb.ro/research/ISOLDE/riscv2ta
UPB	Basic toolchain for the many-core parallel accelerator	https://gitlab.upb.ro/research/ISOLDE/riscv2ta
TUI	Linux Driver SIMD/Vector Accelerator	https://github.com/tui-compute/exp
BYK	Apache StreamPipes Extensions	https://github.com/apache/streampipes

III Future work

Following all the implications of the ISOLDE RISC-V project partners in the virtual repository we have summarized some of the envisioned future work in the following sections.

III.1 Secured RISC-V Processor with AES-256 Cryptographic Accelerator

Predicted Features:

- Dedicated AES-256 hardware accelerator with:
- S-box substitution in LUT or composite field logic
- Key expansion in hardware
- DMA-style memory block handling
- Processor-core integration via:
- Tightly - coupled memory (TCM) or memory-mapped I/O interface
- Custom CSRs or ISA instructions (e.g., aes256.enc, aes256.keyload)
- Verification and security auditing

Future direction includes:

- Formal proofs for AES core (e.g., with JasperGold or Yosys/Symbiyosys)
- Test vectors from NIST SP 800-38A
- Fault injection simulation (e.g., EM fault resistance)

Compiler toolchains will integrate **security-aware optimization levels**, such as:

- `-O2-secure` → avoids spilling keys to stack or unprotected memory
- `-fconstant-time` → enforces constant-time instruction paths

ISOLDE Strategic Edge

Feature	*Value*
Compiler/Accelerator Co-Design	Improves efficiency and code maintainability
AES-256 in Hardware	10–50× performance boost over software
Crypto Hardening + Toolchain Awareness	Enables secure boot, firmware encryption, and communication security
Distribution via Virtual Repository	Immediate reusability for partners and downstream users

III.2 Compiler Support for SIMD & Custom Accelerators

Prediction:

- ISOLDE-specific ISA extension support in compiler backend (GCC/LLVM):
- Pattern matching for AES rounds (SubBytes, MixColumns)
- Auto-vectorization hints for small-scale SIMD (SIMD-lite)
- Integration with RISC-V LLVM TableGen or GCC Machine Descriptions to lower new opcodes.
- Automated offloading support via compiler pragmas:

```c
#pragma riscv_accelerate aes256_encrypt
```

- Support for **LLVM MLIR dialects** or **custom IR passes** to match accelerator behavior (especially if reused in AI/crypto contexts).
- Upstream proposals to RISC-V toolchain projects for reusable cryptographic and vector extensions.

III.3 BAFFI (Bit-accurate FPG fault injector: dependability assessment and verification of FPGA prototypes)

Integration into Dependability Workflows:The ISOLDE RISC-V processor emphasizes **reliability and formal verification**, aligning with BAFFI's goals to assess **soft error effects** and **fault resilience**. Their integration can bridge simulation-level and hardware-level fault assessment.

- **Hardware/Software Co-design Expansion**
  Prediction: BAFFI will be extended to better support **hardware/software co-verification**, where ISOLDE's architecture is tested not just at RTL but within **full-stack applications** running on the core.
  Reason: Fault injection at bit-accurate levels allows for in-depth analysis of how real applications behave under faults — especially useful for safety-critical ISOLDE deployments (e.g., automotive, aerospace).

- **Support for Heterogeneous FPGA Platforms**
  Prediction: BAFFI will evolve to support multiple FPGA platforms and include real-time observability, enhancing compatibility with ISOLDE's prototyping pipelines.
  Reason: ISOLDE cores are often deployed on FPGA platforms for early prototyping. BAFFI's accurate and deterministic fault injection on these platforms would provide a valuable complement to the formal methods used during ISOLDE's development.

- **Enhanced Automation and Toolchain Integration**
  Prediction: BAFFI may integrate with **CI/CD pipelines for hardware** (e.g., using tools like FuseSoC, Cocotb, or Vivado automation scripts) to allow continuous reliability testing of ISOLDE designs.
  Reason: Automation is critical for scalable dependability analysis, especially as RISC-V processors become more customizable.

- **Community & Open-Source Contributions**
  Prediction: As RISC-V and ISOLDE gain popularity, **BAFFI will benefit from increased open-source community engagement**, possibly receiving contributions to expand its injection models, target faults, and performance.
  Reason: The synergy between BAFFI and ISOLDE aligns with the open-source ethos—creating a feedback loop of improvement and adoption in academic and industrial research.

III.4 Toolchain for Hardware-aware Neural Network Optimization

Predictions:

- Introduce compiler passes aware of custom ISOLDE accelerators (e.g., 8-bit MAC blocks, Winograd).
- Graph transformations that map ops to hardware instructions.
- Tool-assisted model slicing and tiling to fit constrained SRAM
- Neural Architecture Search (NAS) loop with MESSY-in-the-loop for hardware feedback.
- Multi-objective optimization: Latency vs. Energy vs. Accuracy.
- Support for tinyML deployment: from model training → profiling → hardware-ready code.

Integration with ISOLDE RISC-V Ecosystem:

- Exports **optimized models and code** into the virtual repository.
- Compatible with **MESSY** for fast simulation-driven design iteration.
- Links with **CI/CD pipelines** to continuously optimize for latest IP changes.
- Tool outputs ready for FPGA validation (bitstream + inference binary).

III.5 MESSY (Multi-layer Extra-functional Simulator in SYstemc)

Predictions:
Layered modeling:

- Cycle-approximate CPU behavior.
- Voltage / frequency scaling.
- Bus contention and memory latency effects.
- Plug-in support for external power estimators (e.g., McPAT, PowerTop-like tools).
- **AI/ML-assisted performance modeling** (data-driven predictions from past simulation runs).
- Integration of **thermal models** (useful for edge devices).
- Scenario-driven simulation (e.g., bursty workloads, failure modes).
- **Open config API** for partners to simulate their custom cores/IPs inside MESSY.

Strategic Value to ISOLDE

Feature	**MESSY**	**TOOLCHAIN**
Hardware-Software Co-design	Models full stack	Adapts ML to silicon
Energy Optimization	Realistic prediction	Compression / quantization
Open Ecosystem	Plug – in IPs	Compatible with ONNX/TF Lite
Reusability	Multi – IP validation	Model re-targeting across cores

IV Conclusions

Through its embrace of the open-source RISC-V instruction set architecture, the ISOLDE project not only advances Europe's digital sovereignty but also fosters innovation in high-performance computing systems. Emphasizing collaboration and open-source hardware, the project is poised to exert a significant influence on European industries and the global technological landscape.