



Network Native

The Next Wave of Connected Embedded Development

Introduction

Virtually all new embedded systems are distributed, heterogeneous designs. With hardware targets ranging from power-sipping 8-bit microcontrollers to lightweight but powerful network gateways to the near limitless resources of internet cloud servers, the modern connected product requires expertise across several disparate development platforms. It's difficult to stay agile while implementing features and fixes across multiple teams, tools, and targets—the logistical challenges slow the pace of product implementation and innovation.

Network Native envisions a new approach to distributed heterogeneous development—one that allows connected embedded components to be developed and maintained as a natural extension of a modern cloud application. Using high-level abstractions modeled on real embedded product use cases, non-specialist developers will discover a new level of productivity and power.

Network Native's product lifecycle ecosystem will enable coding, as well as automated test, build, guided deployment and secure runtime execution for complex heterogeneous distributed systems. **The rapid feature development cycles that have become routine in modern web and mobile development can now be extended to the embedded realm**, decreasing the time required to launch a connected product, introduce new features, and address security and performance issues.

Technology Vision

Sophisticated, agile approaches to building secure cloud-backed web applications have been embraced by developers, but the embedded world has fallen behind.

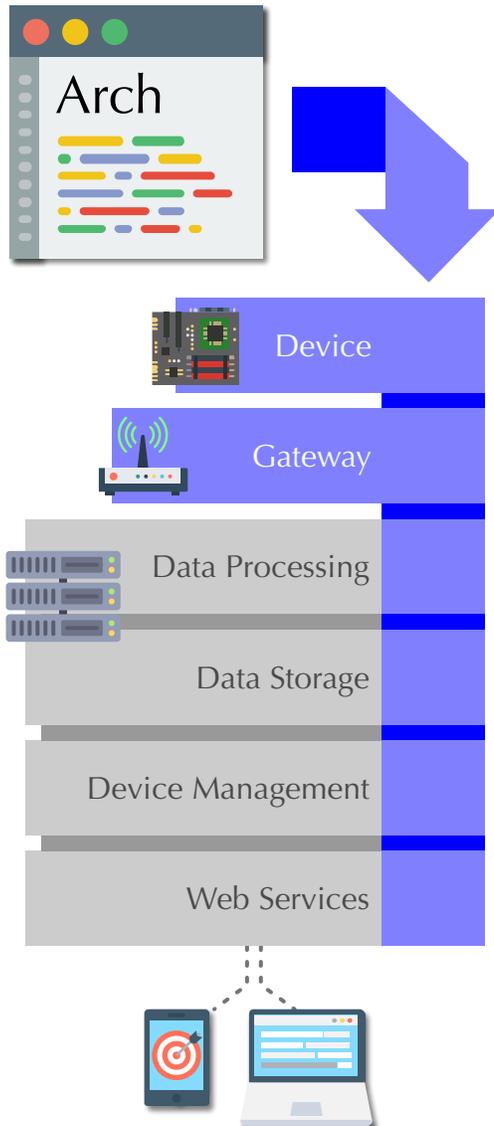
Developing efficient, secure, reliable embedded software still requires highly-specialized knowledge that comes with a steep learning curve. The high-level

abstractions that cloud and application developers take for granted have not found their way to embedded development, making it the slowest, most painful aspect of developing a complete solution.

With a powerful new language, cloud development environment, heterogeneous target compiler, and scalable runtime system, **Network Native will empower developers to design embedded device firmware, secure communications, state management, data processing and storage as a single coherent system.** Instead of dealing with the intense low-level complexity of embedded coding and integration, developers will use high-level abstractions to design from data flow requirements and business rules—finally bringing complete embedded solutions into the modern era of agile development.

Arch Language

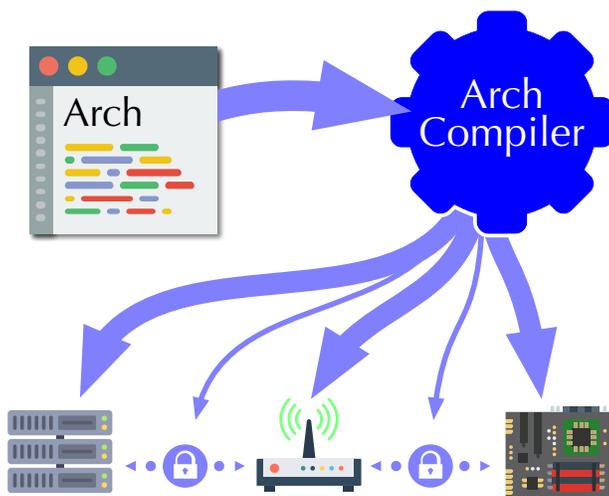
Arch is a new domain-specific programming language designed to allow developers to express event- and data-driven programs using rich abstractions and concise code that can be efficiently compiled for and distributed across a wide range of target hardware.



The Arch language brings together several essential features:

- ▶ **Message-passing primitives** will support proven concurrency and distributed system concepts such as actors, stream processing, and reactive programming. Resulting programs may be split and distributed between processing nodes with little or no developer intervention.

- ▶ **First-class hardware primitives** that encapsulate low-level hardware access, such as bit-oriented registers, memory-mapped I/O, and interrupt callbacks, making them easier for developers to code against, easier to error-check at compile time.
- ▶ **Object-functional design** will support high-level abstractions that greatly simplify the implementation of real world application behaviors.
- ▶ **Whole-program optimization** will enable the generation of small, efficient runtime firmware from high-level code and framework libraries.



Arch Compiler

Building on Arch language message-passing abstractions, **it will become feasible to dynamically and programmatically separate an application into distributable components with transparently-generated secure network communication.** Each distributable component will be emitted as efficient source code for optimized compilation by an appropriate target-specific compiler—leveraging existing embedded toolchains, but without the steep learning curve

and knowledge lock-in.

Framework Libraries

Planned **framework libraries offer efficient, maintained implementations of common software components**, such as:

- ▶ Container data structures
- ▶ Math and data analysis functions
- ▶ Data storage abstractions
- ▶ Protocol stacks
- ▶ Device feature drivers
- ▶ External peripheral drivers

Taking advantage of the heterogeneous design of the Arch language, the framework libraries will offer high-level implementation abstractions while also supporting target-specific specialization that maximizes the efficiency of generated code.

Cloud Development Environment

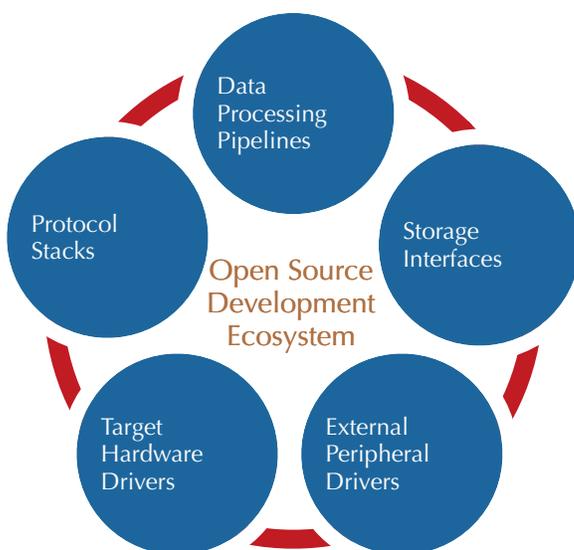
Network Native's **cloud-based development environment will encapsulate all the tools and compilers** necessary to write, test, and build applications using the Arch language and framework libraries, including:

- ▶ **Service APIs** — incorporating source code analysis, testing, compiling, and building—enabling integration with client-side IDE plugins and build systems
- ▶ **Automated test and build** — with cloud-based native embedded compilers
- ▶ **Web-based IDE** — source code browser, version control tool, intelligent source code editor, test runner, and compilation tool—built on top of the Service APIs

Deployment

The **cloud-based deployment system will coordinate transactional deployment of new application components** to all appropriate target nodes through the best available deployment mechanism for each target. Support for multiple concurrent running software versions will ensure maximum uptime during partial or staged deployment scenarios.

Service APIs will allow deployment to be tied into existing software deployment workflows.



Community Vision

The open-source process has proven itself in countless high-value production-ready software systems. Capturing all the value of open-source development in the embedded realm has been difficult because portability and ease of use have often been traded for runtime performance and efficiency. Combined with proprietary development environments, the result is reusable code and production-ready libraries tend to accumulate only around particular vendors or families of chips.

Network Native plans to realize the power of high-quality, portable, maintained code in embedded development by combining technical and administrative approaches.

Through the **flexibility of the Arch language**, a shared heterogeneous codebase will be feasible for a wide range of target platforms; developers can be assured that their contributions are benefitting the whole ecosystem.

Some aspects of the open-source process might seem to be at odds with the rigorous performance and validation requirements of the embedded software world. The Network Native approach is to have **strongly managed code acceptance, validation, and versioning processes** and to ensure visibility into the provenance and maintenance of the software ecosystem.

Commercial Opportunity

Silicon vendors who embrace this movement will reap the benefits of offering their customers a powerful new development process that will **shorten product development cycles**, promote **design wins based on price/performance** instead of toolchain lock-in, and allow **hardware innovations to be immediately adopted by customers**.

Since Network Native will incorporate existing native toolchains, current **organizational expertise will be retained and can focus on increasing product capabilities instead of replicating common tools and user interfaces**. By eliminating the requirement for customers to have extensive low-level hardware knowledge, transitioning to new or unique existing cores will be greatly simplified—**giving new life to 8- and 16-bit cores** that otherwise might be bypassed by developers accustomed to the ease of a 32-bit ARM target.

Join Us

Network Native is pioneering the next wave of distributed heterogeneous software development. The Arch language, framework libraries, and compiler are in the process of becoming an open-source community project. Network Native seeks industry and academic partners interested in participating in the **Arch Development Consortium**.

Network Native

<http://networknative.com>

join@networknative.com