Elettra Sincrotrone Trieste

Voltumna Linux: a custom distribution for (embedded) systems

THPDP026

L. Pivetta, A.I. Bogani, G. Scalamera – Elettra Sincrotrone Trieste

In the last years a thorough approach has been adopted to address the ageing and the variability of control system platforms at Elettra Sincrotrone Trieste. The second generation of an in-house built operating system, named Voltumna Linux, which is based on an immutable image approach, is now ready for production, supporting a number of commercial-off-the-shelf embedded systems. Moreover, the same approach is perfectly suitable for rack-mount servers, with large memory support, that often require the inclusion of third party or closed source packages. Being entirely based on Git for revision control, Voltumna Linux brings in a number of advantages, such as reproducibility of the product, ease of upgrading or downgrading complete systems, centralised management and deployment of the user software to name a few.

1 - INTRODUCTION

In recent years the number of front-end machines used within Elettra control systems has increades considerably. Legacy systems, such as VME single board computers based on MC680x0 and PowerPC microprocessors, were joined by x86 systems in standard 19 inch form factor, like rack-mount servers, and smaller form factor such as NUC, UPBoard, Jetway and MinnowBoard. Moreover, also embedded boards based on ARM microprocessor, such as the Beaglebone, and system-on-chip boards based on FPGA, e.g. De10-Nano, Sockit, Dinet and Arria10 have been adopted. At Elettra, control systems are mostly based on GNU/Linux distributions, adopted over the years, many times featuring hard real-time extensions such as RTAI [1] or Xenomai [2]. Keeping the same approach for new platforms, over the time, would lead to a even more heterogeneous install base, with additional GNU/Linux distributions or versions. Moreover, together with the new platforms, existing systems have to be maintained much longer than the typical commercial distribution support, which several times is limited to security updates.

2 - REQUIREMENTS

1. Allow the adoption of specific versions of system components
2. Allow to integrate third party software, when source code is available
3. Provide multiple levels of customization (kernel, drivers, libraries) by patching or revision control
4. Optimise the operating system for each hardware. 5. Guarantee reproducibility, for both the operating
system and the BIOS/firmware of motherboard and adapters
6. Build, whenever possible, system configurations first
7. Encourage software reuse making it available from initial releases
8. Minimise platform dissimilarity, with special attention
to operating system and low level software stack
9. Provide separate images for development and production systems
10. Simplify working with new or low performance platforms supporting cross-compiling.

3 - YOCTO / OPENEMBEDDED

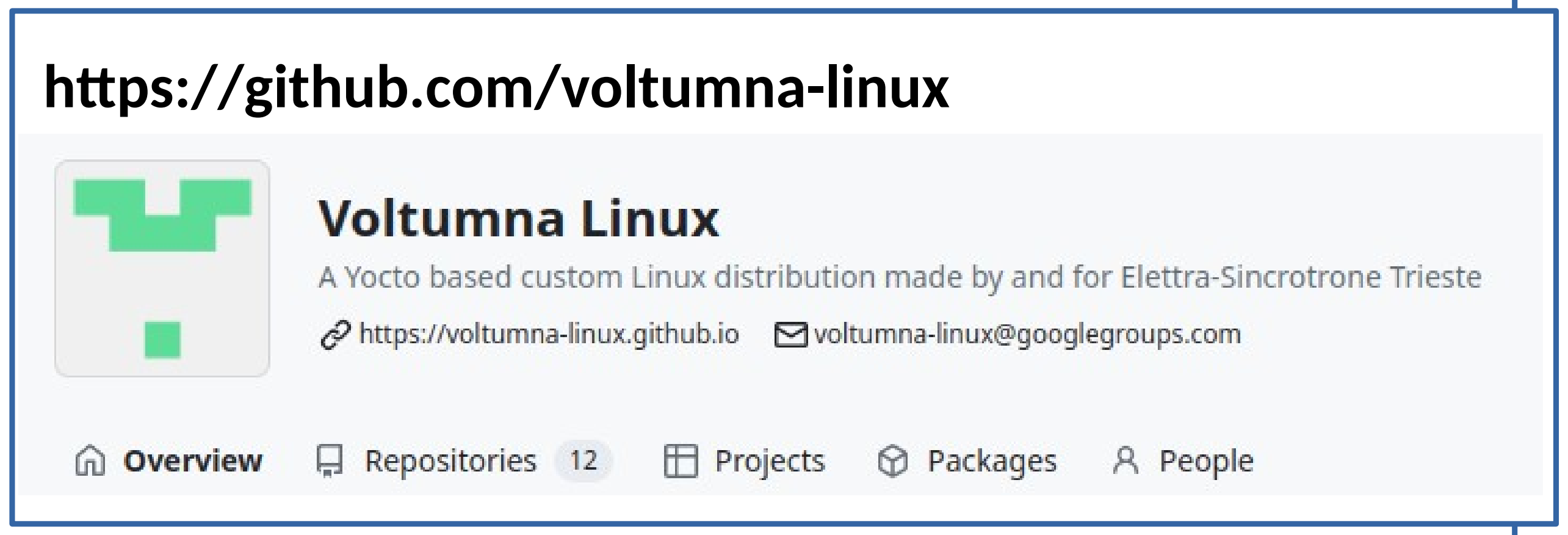
The Yocto/OpenEmbedded Project [3] is an open source collaboration that provides a flexible set of tools to create custom GNU/Linux based systems for embedded hardware, regardless of the architecture. Established in 2010, it involves many hardware manufacturers, including AMD, ARM, Intel, Texas Instruments to name a few, open- source operating system vendors and electronics companies. Within the boards in use at Elettra, Yocto/OpenEmbedded is supported by Terasic for the Sockit system-on-chip FPGA based board and by Texas Instruments for the Beaglebone. Therefore Yocto/OpenEmbedded was the natural, more convenient, choice to base a new, custom, GNU/Linux distribution to be developed in-house.

4 – VOLTUMNA LINUX

Voltumna, is based on the concept of meta-distribution, defining a base system that can be kept as is or customised in additional steps. This is done properly aggregating several, pre-existing or custom built, layers. Each layer defines a collection of text files, used to generate or modify a software package or its configuration, named recipes.

When needed, a recipe belonging to a lower layer can be customised within a distribution. Third party applications, requiring a specific kernel release, or some specific version of system libraries, are typical use case. Third party applications can be included in Voltumna in both source code or binary format, whether in-house software is always included in source code, and compiled when generating the distribution image.

This approach allowed to adopt the same hardware, a power supply designed by Elettra, and integrate the support for different facilities: the layer meta-elettra for use at Elettra and the layer meta-ess for use at ESS. All the underlying layers, that constitute the larger part of the system, are exactly the same.



https://github.com/voltumna-linux

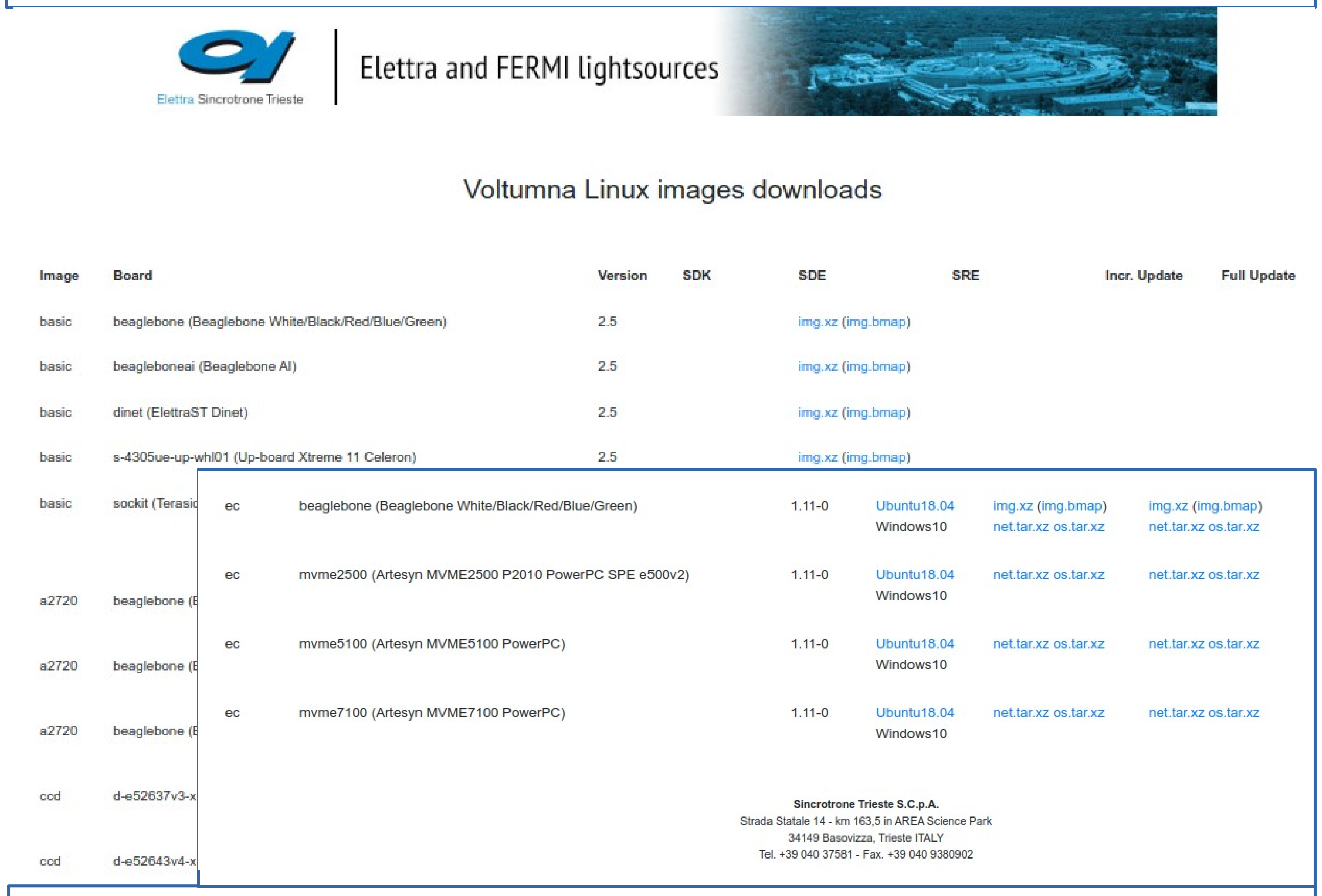
Voltumna Linux

A Yocto based custom Linux distribution made by and for Elettra-Sincrotrone Trieste

https://voltumna-linux.github.io voltumna-linux@googlegroups.com

Within Yocto/OpenEmbedded each platform is represented by the MACHINE file abstraction. Voltumna defines its own MACHINE file for each specific target; this allows, for instance, to **build the same distribution for different platforms while heavily optimizing the kernel for the instruction set of each CPU**.

Voltumna linux distribution images are available in two flavors, based on the destination use, and a software development kit for each target. The Software Runtime Environment (SRE) image is finalised for deployment on production hosts, where system resilience and reproducibility are fundamental. With these goals in mind, the SRE is built by essential components, reduced to the minimum, and totally lacks any development tool or documentation. The Software Development Environment (SDE) image provides a development environment, complete with all development and debugging tools, built to run on the target platform. The Software Development Kit (SDK) is an installable package, with the cross-compiler and the development environment proper to the target system.



Voltumna Linux images downloads

Image	Board	Version	SDK	SDE	SRE	Incr. Update	Full Update
basic	beaglebone (Beaglebone White/Black/Red/Blue/Green)	2.5		img.xz (img.bmap)			
basic	beagleboneai (Beaglebone AI)	2.5		img.xz (img.bmap)			
basic	dinet (ElettraST Dinet)	2.5		img.xz (img.bmap)			
basic	s-4305ue-up-whl01 (Up-board Xtreme 11 Celeron)	2.5		img.xz (img.bmap)			
basic	sockit (Terasic						
a2720	beaglebone (E						
a2720	beaglebone (E						
a2720	beaglebone (E						
ccd	d-e52637v3-x						
ccd	d-e52643v4-x						

ec	beaglebone (Beaglebone White/Black/Red/Blue/Green)	1.11-0	Ubuntu18.04 Windows10	img.xz (img.bmap) net.tar.xz os.tar.xz	img.xz (img.bmap) net.tar.xz os.tar.xz
ec	mvme2500 (Artesyn MVME2500 P2010 PowerPC SPE e500v2)	1.11-0	Ubuntu18.04 Windows10	net.tar.xz os.tar.xz	net.tar.xz os.tar.xz
ec	mvme5100 (Artesyn MVME5100 PowerPC)	1.11-0	Ubuntu18.04 Windows10	net.tar.xz os.tar.xz	net.tar.xz os.tar.xz
ec	mvme7100 (Artesyn MVME7100 PowerPC)	1.11-0	Ubuntu18.04 Windows10	net.tar.xz os.tar.xz	net.tar.xz os.tar.xz

Sincrotrone Trieste S.C.p.A.
Strada Statale 14 - km 163,5 in AREA Science Park
34149 Basovizza, Trieste ITALY
Tel. +39 040 37581 - Fax. +39 040 9380902

5 - DEPLOYMENT

Typical deployment for front-end computers is based on **network boot and diskless operation**. Each facility deserves at least one server for network services, one for shared filesystem, based on NFS, one for the Tango database, all deployed as virtual machines, plus a number of physical and virtual hosts running the controls system devices. Voltumna Linux is based on **immutable image approach, regardless the target is booting over the network or from local storage**: the main system folder, where Voltumna is stored, is mounted read-only. When in diskless operation, target systems mount a private root filesystem in read-write mode, but everything under /usr is shared, reducing storage requirements on the NFS server as side benefit. The Software Development Kits for Voltumna Linux distributions, supporting the target systems, have been integrated into Elettra's automatic build and installation system, named INAU [8], and are available to the developers like legacy development environments.

[1] RTAI, https://www.rtai.org/
[2] Xenomai, https://www.xenomai.org/
[3] Yocto Project, https://www.yoctoproject.org/
[8] L. Pivetta, A.I. Bogani, "INAU: a custom build-and-deploy tool based on Git", PCaPAC 2022, Dolní-Břežani, Prague, THP01

Elettra-Sincrotrone Trieste S.C.p.A
Strada Statale 14 – Km 163.5 in Area Science Park
34149 Basovizza, Trieste ITALY

ICALEPCS CAPE TOWN SOUTH AFRICA 2023

lorenzo.pivetta@elettra.eu
alessio.bogani@elettra.eu
graziano.scalamera@elettra.eu