

- PRESS RELEASE -



**METISMO ANNOUNCES 'BEDROCK' CROSS COMPILER TECHNOLOGY
FOR ANTIX GAME PLAYER**

**- Enables Java and DoJa games to be deployed easily across Antix-
ready devices -**

23rd June 2008: Metismo and Antix Labs today announce a collaboration to enable wider mobile games deployment opportunities for publishers of games originally written in Java. Metismo has added Antix Game Player support to its 'Bedrock' Cross Compiler tool, allowing its licensees to quickly translate Java-based games content into C++ which can be deployed via the Antix Game Player across Antix-enabled devices, reducing porting costs and accelerating time-to-market.

Through this co-operation, Java games developers and publishers are now able to target Antix-enabled platforms with native games. The games' native nature allows them to run more efficiently, improving CPU usage and battery life. Furthermore, fragmentation is reduced and with no need to write different ports of games, publishers can divert more attention and investment to development of the game itself, rather than re-coding.

John Chasey, Chief Executive at Metismo, comments, "Antix has been gaining significant traction with device manufacturers and accordingly we've seen an increasing demand from publishers to target the Antix Game Player platform. We can now offer this service to publishers, further reducing the number of SKUs required for a wider range of devices."

Francis Charig, Chief Executive of Antix Labs, comments, "Metismo's Bedrock solution allows Java developers to exploit a powerful enabling technology benefiting from deployment across different devices with just a single executable. The optimum way publishers, manufacturers and network operators alike can truly maximize the latent demand from consumers for mobile gaming is to adopt a single strategy that will allow games to be simultaneously deployed across many platforms. This collaboration provides an environment that reaches far across the increasing wealth of different multimedia devices."

Using techniques never used before in released gaming products, content written in C and C++ on the Antix Game Player runs stunningly fast across Antix-enabled targets while remaining independent of software platform and hardware. With the ability to run across multiple open and proprietary platforms, including entry-level and high-end handsets, the Antix Game Player uniquely provides a single, advanced, industry standard development tool chain for all platforms supporting Antix. The game binaries developed on a PC are identical to the game binaries on the target.

- Ends -

About Metismo

Metismo is a mobile middleware company and a creator of mobile applications. Metismo makes robust tools and systems for mobile application development and deployment, and are also working on innovative consumer products that will let you take your mobile device to new heights.

For further information, please contact:

John Chasey

Metismo Ltd

Tel: +44 (0)7967 656952

Email: enquiries@metismo.com

Web: <http://www.metismo.com>

About Antix Labs

Antix, the innovative, compact and powerful, universal player running single SKU, 2D and 3D, P2P games on phones, televisions and personal computers, accelerating content time-to-market, reducing development costs, exploiting existing assets, improving content discovery and designed to meet the consumer's ever increasing expectations and aspirations.

Antix Labs, headquartered in the UK, was founded in 2007 by Francis Charig, a World Economic Forum Technology Pioneer. The Antix team has accumulated more than 500 man-years experience in the design and bringing to market of rich media software platforms working for operators, handset manufacturers, games publishers, compiler companies and silicon design houses.

For further information, please contact:

Neil Curtis
Antix Labs

Tel: +44 (0) 118 357 0 357

Fax: +44 (0) 118 357 0 358

Email: neil.curtis@antixlabs.com

Web: <http://antixlabs.com>

Will Gardiner
itpr

Tel: + 44 (0) 1932 578 800

Fax: + 44 (0) 1932 578 801

Email: willg@itpr.co.uk