



mophun™ - Real games in your mobile phone



White Paper

May 2002

Contents

Preface	3
Introduction	3
End user Experience	3
mophun technology	4
mophun RTE	4
gaming API	4
Security system	4
SDK	4
Developing mophun games	5
mophun.com developers community, main point of contact	5
Development with the mophun SDK	6
Gamelet certification	6
Business model	7
Revenue model	7
Provisioning	7
The Loader gamelet	7
Standard OTA provisioning	8

Preface

The purpose of this document is to give the reader a general understanding of mophun™. People who can benefit from this document are operators, content aggregators, publishers, developers and business decision-makers.

Introduction

mophun™ is a software based gaming console for mobile terminals. A state of the art gaming experience requires performance only possible with optimized graphics and interaction access that the mophun™ platform has perfected.

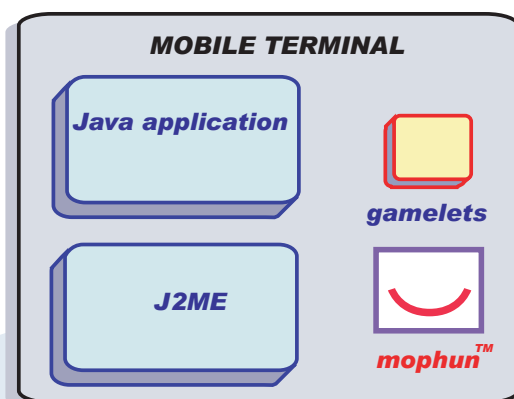


Figure 1. mophun™ can coexist with J2ME: mophun™ is for games, J2ME is for everything else

mophun™ gamelets are the killer applications that will drive traffic and revenue for operators. Considering the popularity of multi-player and location based mobile games today the downloadable rich clients that mophun™ enables will revolutionize mobile gaming.

The image below describes the main actors involved in the distribution of games for mophun enabled phones.

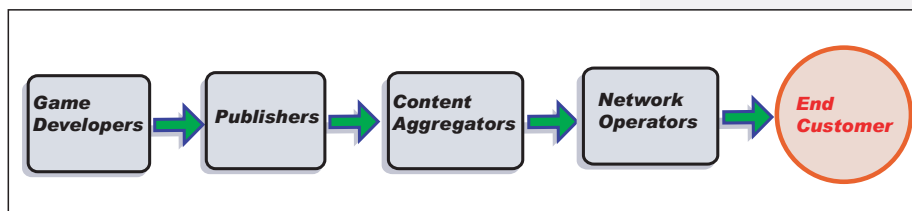


Figure 2. The value chain

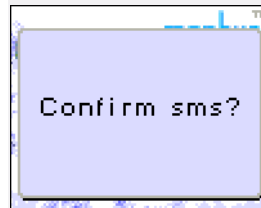
End user Experience

This is what it looks like for the end user to download and launch a mophun game.

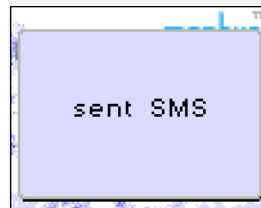
1. Select the game to download



2. Confirm sending the SMS request



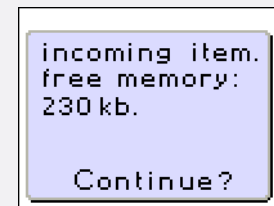
3. Request has been sent



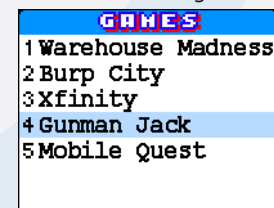
4. File transfer



5. Confirm the incoming push download



6. Navigate to the games menu and launch the new game



7. Play !!!



mophun technology

mophun RTE

The mophun™ runtime environment consists of an ultra-slim virtual machine, requiring less than 10 kb of memory for execution. Because of the small size it can fit completely in the instruction cache (if it exists) of the host CPU. It is equipped with a super-efficient native code interface requiring only about 10 machine instructions (depending on the host system) overhead for calls to native functions.



gaming API

The gaming API provides all the functionality a game requires like platform independent video access, input, sound and communication. A query interface makes it possible to adapt a game to the capabilities of the device. The API is 100% native code to deliver the best performance possible. Unlike other mobile platforms mophun™ is optimized for writing games. Some differentiators are a built in sprite engine, map engine, different color depth tiles, transparency support etc.

To take advantage of the varying capabilities of different platforms, code and data can be isolated into modules that are loaded according to these capabilities. For example, it is possible to have both grayscale and full color graphics resources but only load the specific resources that are compatible with the surrounding system. If a game does not provide multiple resources, the mophun™ system handles the conversions automatically.

Future extensions include for example a 3D API.

Security system

A mophun™ gamelet is fully verified and executed in a secure sandbox environment to prevent a program from attempting illegal actions. All interactions with the outside world is handled by a secure layer in between the mophun™ application and the "real" system interfaces.

To further enhance security mophun™ gamelets need to be digitally signed to be allowed to be executed on the mobile terminal. Gamelets are signed in the publishing process after being screened for malicious content.

SDK

The mophun™ SDK is a non-proprietary completely free toolset built with GNU tools using open standards. It was built by game developers for game developers. Games are written in C/C++ or assembler, the preferred programming languages for games programmers. The software can be written and tested completely on a regular PC without having access to a real device. Source level debugging is available for both the Linux and Windows versions of the SDK. The SDK even includes emulation profiles for the target mophun™ enabled mobile terminals. It also ships with a full set of binary utilities for manipulation and handling of code, data and resources.

Developing mophun games

This is a generic description of the entire process involved in the release of mophun games, from development to distribution.

mophun.com developers community, main point of contact

We have started www.mophun.com as the main point of contact for mophun developers, mophun.com has started as a forum and will shortly evolve into a full blown developers community, although the main features are already available. You will here find an answer to most of the questions related to mophun, and the possibility to communicate directly with mophun developers. The following functionality is available at mophun.com:

member updates

To have access to the SDK and development tools the developer must be a registered member. Only registered users will get updated on news, product updates, upcoming events, etc.

news

All latest news (development and business related) are exposed in the mophun community.



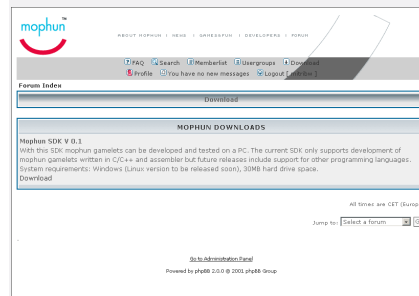
games

The latest certified games will be exposed in this area after their release.



downloads

Members will be able to download the latest SDK and other development tools.



forum

The forum is the main point of contact for mophun developers. A number of discussion topics are initially proposed but the goal is to have the forum administered by the members of the mophun community.

Forum	Topics	Posts	Last Post
News			
mophun news	2	2	Thu May 24, 2002 10:27 pm
Development			
Newbie	0	0	No Posts
General	9	57	Thu Apr 25, 2002 4:17 pm
Tips and tricks	1	2	Wed Apr 24, 2002 9:14 pm
Bugs	0	0	No Posts
3rd party Software	0	0	No Posts
GAMES			
Talk about Games	0	0	No Posts
Misc			
General	2	7	Wed Apr 10, 2002 6:10 am
Suggestions	0	0	No Posts
Runtime			
windows	1	1	Sun Apr 14, 2002 3:50 am

Some comments from the mophun community

hey i got the beta sdk and everythings soo good.
took me minutes to get apps building for it.
does anyone want to help me start a dev site for mophun, so people can upload there source and proggys and we could have compertitions and that.
i can make the whole site my self but wondered if ppl wanted to help me make a good one :)

Development with the mophun SDK

The process of developing a mophun game is very straightforward. In order to quickly introduce the programmer to the mophun functionality and API there are several simple tutorials.

The programmer has the possibility to run everything from the command line right out of the box but there are many other ways to set up the development environment for mophun™. The programming guidelines show how to use Microsoft Visual Studio™.

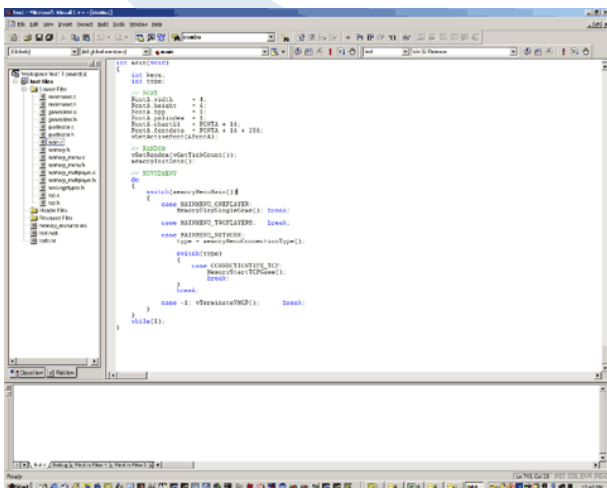
Runtime environment

Just drop the mophun compiled game into the mophun runtime. It emulates the performance on the target device.



Development project

The development of mophun games is very much like development of traditional console based games



Gamelet certification

Certification is an essential step in order to preserve the security aspects on mobile terminals. It is also a requirement from phone manufacturers to prevent offensive content in the phones.

Game idea submission

Initially and in conjunction with the release of the first enabled mophun phones, developers will have to submit their game ideas. Those are evaluated and may be part of the initial launch. This is to secure that high quality content is produced. With time this role could be taken over by content providers/aggregators and publishers.

The following will be a part of the game idea submission process until approval.

You must be a registered member of the mophun community.

Send complete company information/presentation.

Send your game concept document and game design document.

We advice you to send a demo (Download the SDK from our site)

All the information will be reviewed and we will get back to you with a response.

Description of your preferred business model, how do you plan to get paid for the game, i.e will you implement a subscription model, pay/download, first level free and then pay for each level?

If necessary we will get back to the developer with necessary adjustments.

Certification process

Gamelets must be certified to be allowed to run on a mobile phone. Certifying a gamelet means to quality assure it. We make sure there is no offensive content, that the gameplay and functionality is according to settled standards, etc. We also scan the gamelet for possible security breaches. If the gamelet passes the certification process, the gamelets are signed and returned to the developer. Now the gamelets are ready for their distribution. Remember, no changes can now be done on the signed gamelet.

Business model

So how does it work? Who will make money out of this? Once a game has been produced on the mophun™ platform it will be certified by Synergenix (described above). The certified game may be distributed and sold by companies who specialise in game distribution, to operators who in turn will sell the games to the end users or through their own distribution channels.

Let us take an example. Developer X produces a new mophun game. He then handles it to us and it gets certified and thereby signed and returned to the developer. The game is ready to sell. He has now different choices available. Either:

He contacts a content aggregator to distribute the game. The aggregator has a network of operators and will assure the global distribution of the game.

He contact his publisher and thy are handling the distribution.

He may take direct contact with the operator/s.

Either way the games will find their way to the end users.

It has been estimated that every fourth mobile phone user will download games and they are willing to pay approximately 2-3€ per game each month. Take that number and multiply by the number of phones expected to be sold by next year and you see that there is quite a bit of money to make. The 2-3€ should be split between the game distributor and the game developer after that Synergenix has charged for a licensing/certification fee.

Revenue model

We have implemented an Open solution for developers to introduce the revenue model they estimate most appropriate. It is up to the developer (in cooperation with Content provider, Aggregator, Publisher or Operator) to design the revenue model that best suits he's needs. Thereby the distribution of the game might be by pay/download, Pay/level, pay/Credits etc. In the end, revenues are shared between the involved parties (see value chain).

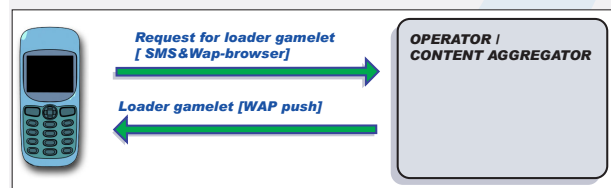
Synergenix is a technology enabler. We provide content aggregators and operators with a secure solution to keep track of the number of files that are downloaded. A file can consist for example of a new game, credits or new levels. The game developer decides if a "credit file" should consist of 10, 50, 100 or any other number of credits or if a "level file" should consist of 1, 5, 10 or any other number of levels.

Provisioning

On java enabled phones standard provisioning is supported. The solution on non java enabled phomes is described below.

mophun™ OTA (Over The Air) provisioning is based on standard WAP push and Premium SMS. It is not dependent on GPRS access.

The Loader gamelet



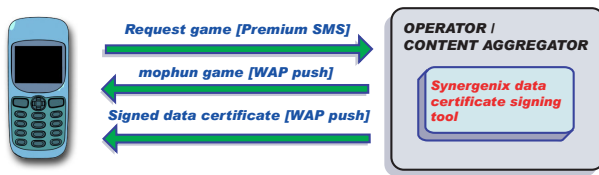
The standard scenario for mophun provisioning is using a loader gamelet. A loader gamelet is a special gamelet that is provided by the content aggregator/operator. The "game" is basically an SMS extension that lists the gamelets available on the server. The end user can browse through the list and select the game he wishes to download. In the provisioning described below it is assumed that a loader gamelet is already in the mobile terminal, which can happen in the following ways:

The content aggregator/operator pushes its loader gamelet to its customers via a WAP push link. The end users accept the link and download the loader gamelet at no cost.

The end users initiate the request for a loader gamelet by sending an SMS or visiting a wap portal. The content aggregator/operator proceeds as above.

The mobile terminal retailers bundle a loader gamelet with the mobile terminal at presale.

Standard OTA provisioning



When the end user selects a game the loader gamelet generates an SMS that contains the game name and a unique ID for the mobile terminal. The Synergenix data certificate signing tool is used to digitally sign a data certificate so that it is only valid on the mobile terminal that originated the request.

Viral marketing

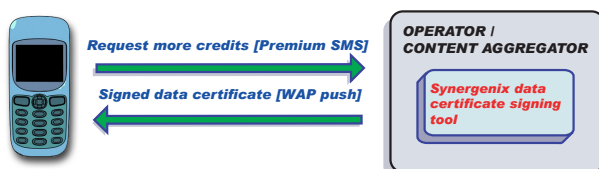
If the gamelet is sent via beaming to another user it could be designed as a demo game and the new user has to buy his own credits to continue playing or get access to the full features of the game.

Pay per download

In the scenario above the game is a playable demo by itself and the data certificate enables the full functionality of the game.

Pay per play

Most games will probably not charge for every single play but rather charge for a number of credits, lives or extra levels etc. Regardless of the type of digital right to play it's still basically a pay per play scenario where the end user generates a buy request SMS from within the game and a new signed data certificate is returned:



Forwardlock

In the above examples the signed data certificates are only valid on the requesting mobile terminal so beaming them is pointless as it will be invalid. On top of this commercial games can also have extra protection using an encrypted ForwardLock setting that disables forwarding via beaming to other users.

Promotion games

It is possible to have non-commercial promotion games that don't restrict access to features of the game via signed data certificates. In this case the Synergenix data certificate signing tool is not used and the game is pushed back without using any data certificates.

However, unsigned games will not have communication or file access and might be restricted in size as well. The reason for this is to protect revenues for commercial games.

Since promotion games don't require a billing system, detection can be done with a WAP browser but for consistency operators/content aggregators may choose to use the loader gamelet and SMS scheme in this case as well.

