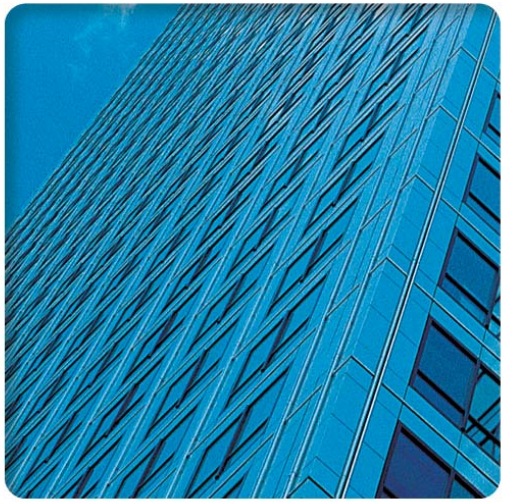


Engineering with a Business Plan

(parts of a starting up guide :-)

John Pagonis
Symbian Developer Network

This talk

- Address the apparent opportunity of entering the mobile phone applications market as an ISV
- Starting up in this business
- Give you some pointers on sourcing information about the mobile phone applications business
- Discuss the product sales cycle
- Briefly touch on the issues of distribution

Why am I doing this ?

- Help developers develop
- …and we need them to be in business
- …and keep developing for Symbian OS

There is no point for Symbian to attract developers into a situation where they will fail !

Better **not** come along if you cannot make it.

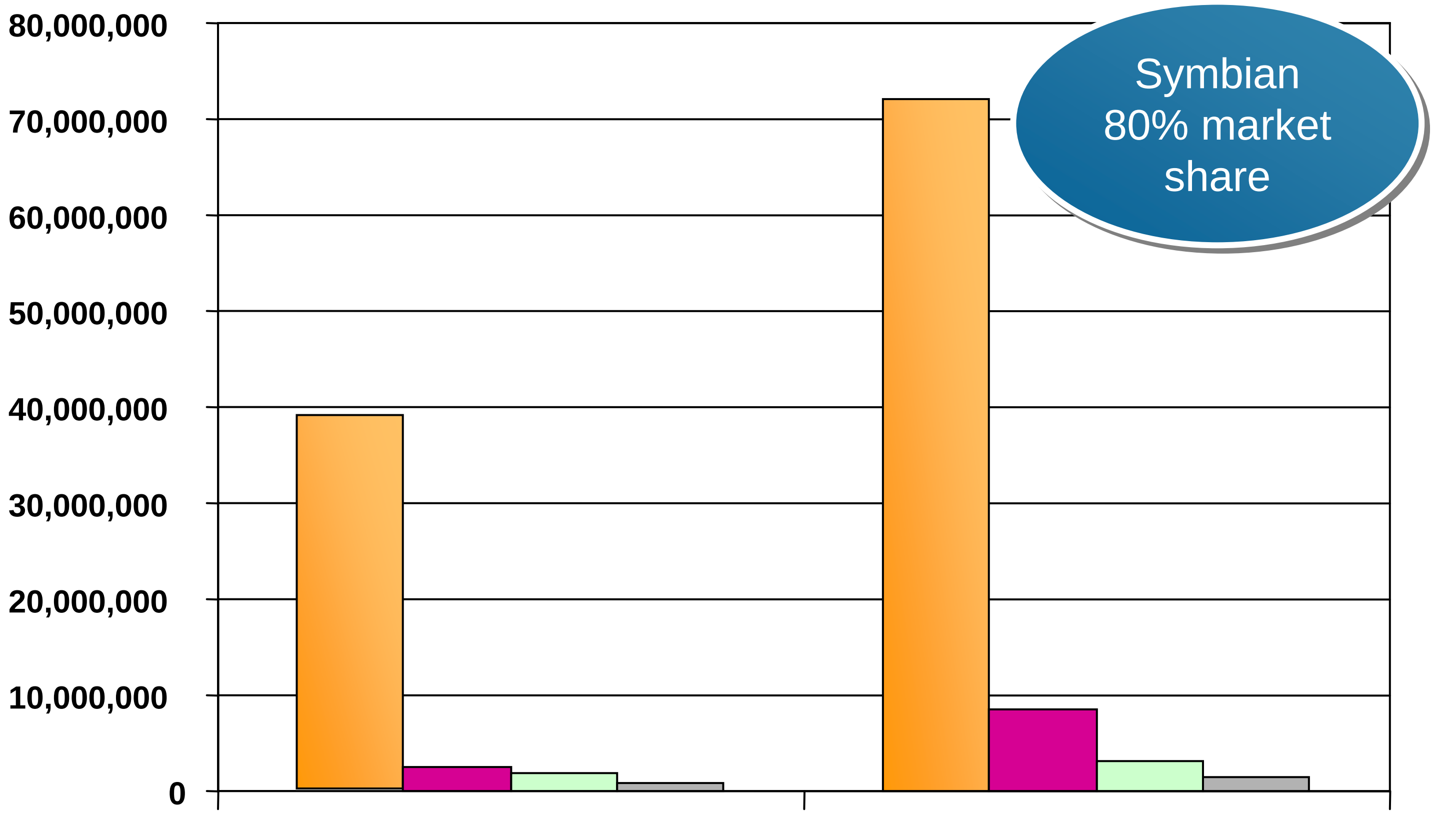
Dead developers are no good to anyone’s business

Marketing and Engineering

- You need both (and much more)
- I’ll be talking about both, be prepared
- ‘’technology, usability and marketing’’ as D. Norman puts it (‘’The Invisible Computer’, MIT Press)

Mandatory graphs

Source: Canalys

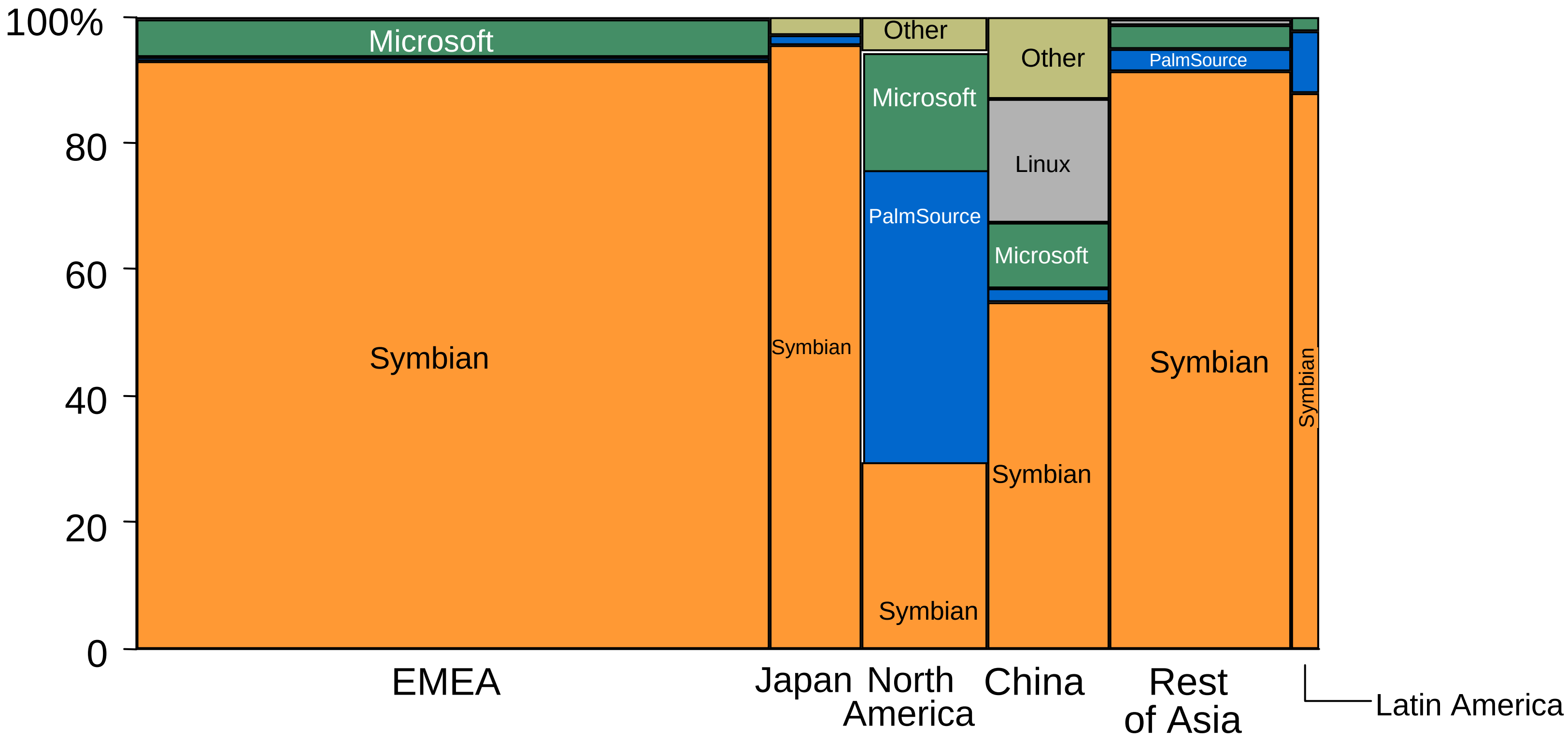


Symbian 80% market share

Regional smartphone OS market shares

Canalys

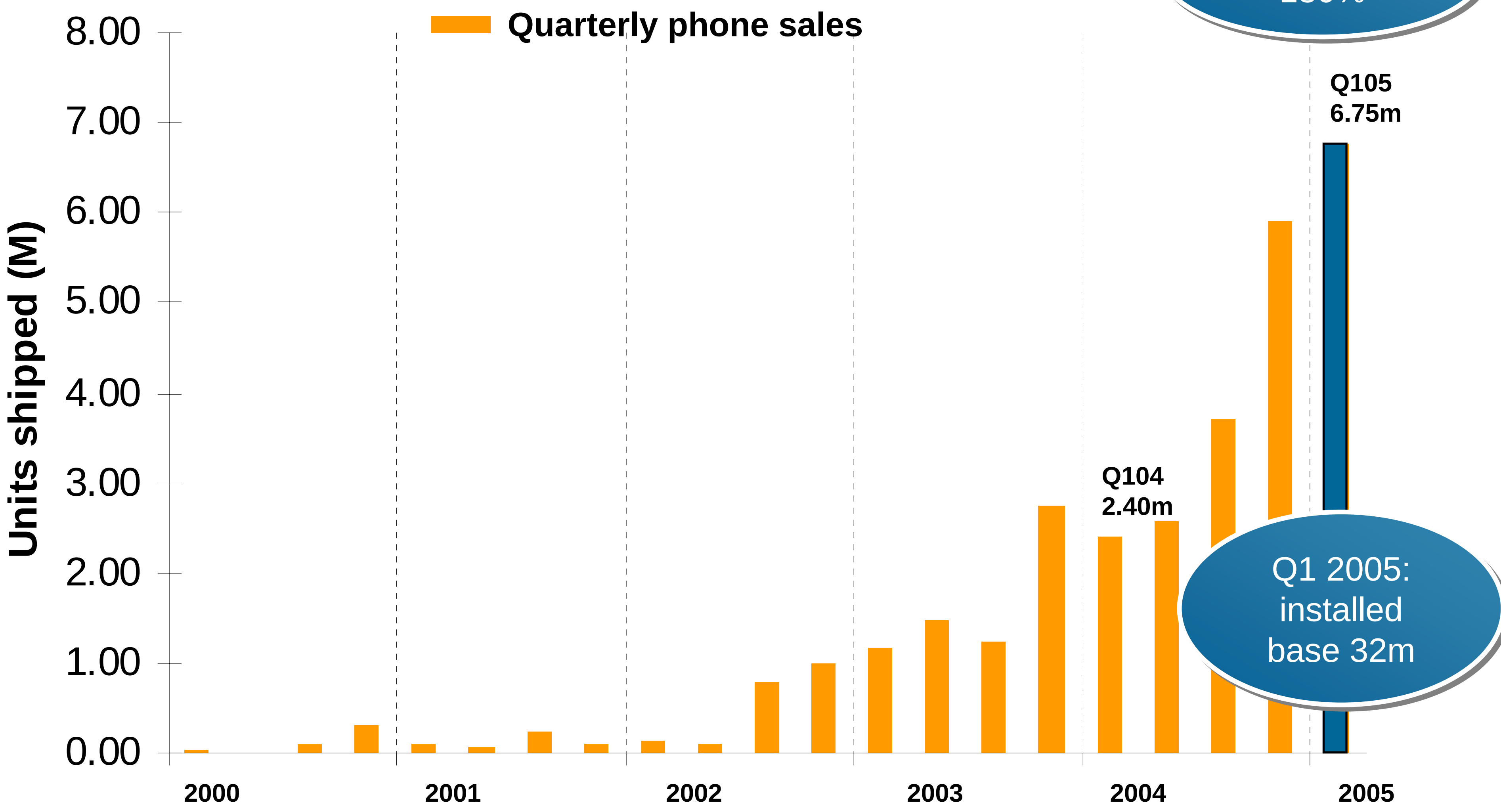
2004 smartphone unit sales by region by OS vendor



Source: Canalys, January 2005

Smartphone is defined as voice-centric device running open OS (e.g. excl. Blackberry or XDA)

Symbian OS-based phone shipments



YoY growth 180%

Q1 2005: installed base 32m

Starting up….(an oversimplified naïve view)

- Opportunity
- Spotting the need(s)
- Building the team
- Building the infrastructure
- Creating the product
- Marketing the product
- …and continue being in business

So what is the opportunity ?

- ✓ An installed base of ~40 mil. Symbian OS phones ?
- ✓ Because these numbers keep growing ?
- ✓ Around 80+% of ‘’the smartphone market’’ ? (Canalys report, ‘Mobile Device Trends 2005’)
- ✓ Low barriers to entry ?
- ✓ New ‘’big thing’’ ?
- ✓ The next enterprise frontier ?
- ✓ Innovative services ?
- ✓ Nascent market, ready to be harnessed ?
- ✓ Unfulfilled user needs ?

But how much do we know about this market ?

- Who is buying software for their mobile phone?
- Why, how, when, where ?
- What do people buy ?
- What do people need ?
- What is the product sales cycle ?
- How about pricing ?
- How about segmentation ?
- What is the addressable market in every segment ?
- What will it be in 3 years ?
- Influencing trends ?

Where can we get more information about patterns in this market ?

Profound statement… (keeps you awake:-)

As far as the independent mobile phone software business is concerned, we are circa early 80s !

- ❑We still have to get a Visicalc (a.k.a. software that makes you want to buy a particular phone)
- ❑Nobody has sold as much as the ZX-81/Spectrum Flight Simulator yet (Psion had sold more than 1 million copies, by 1982)
- ❑Most predictions will be wrong (''who needs more than 640K'' etc)
- ❑It is going to be disruptive

Profound statement continued….

As far as the independent mobile phone software business is concerned, we are circa early 80s !

But this time around we have:

✓More knowledge (in both engineering and marketing)

✓The Internet (if you were on the internet back then, I salute you:-)

✓More users (a.k.a. potential customers) already

✓Better infrastructure and tools

Now, there is an open platform on which to work on :

Symbian OS

Here lies the Opportunity

As far as the independent mobile phone software business is concerned, we are circa early 80s !

- Who wouldn’t want to be the software business’ who got it right back then?
- Things are happening much faster than back then, agility is as important as strategy and execution

Q: So would you want to miss out ?

A: No. that is why we are here today :-)

So you want to do it then ?

- I’m afraid it is not for everyone (competent team)

- It is not easy (skilled and seasoned team)

- Still cheap to enter (small team or homebrew development)

Homebrew development

‘Return of the homebrew coder’, The Economist, March 11th 2004

‘’ Most modern software is written by huge teams of programmers. But there is still room for homebrew coders, at least in some unusual niches’’

There are 3.8 million internet home workers in the UK, Broadband User Survey by Point Topic, Sept 2005

Broadband adoption and free Internet based infrastructure (IM, VoIP, SVN) also allows for low cost dispersed team development…

As far as building the team goes…

- The most difficult and expensive ‘’task’’ in software engineering is to build ‘’the team’’.
- Whether in the same room (recommended) or dispersed, the same organisational patterns apply (‘Organizational Patterns of Agile Software Development’, by J. O Coplen and N. B Harisson).
- Make sure the team has domain specific knowledge and does appreciate that developing for a mobile environment ***is*** different.
- Dispersed development can be much cheaper, but it is hard

About language selection

- The trade-offs between Symbian OS C++ and Java is not only of technical skills available but also a business one.
- Available functionality vs skills vs TTM
- Java doesn’t always imply better TTM
- You may have to wait for a JSR to be realised and deployed on mobiles, while the competition delivers in C++
- But you’d need to train people to Symbian OS C++ before you can deliver.
- …there lies one of the big trade-offs

What’s next ?

- So you have the team
- You see the opportunity
- You found the customers(?)
- You have spotted what your customers need(?)
- The investment seems low enough

…now you need to budget !

R&D cost base, pricing, net price, distribution, promotion, infrastructure, *sales cycle* …

To budget you need to know the sales cycle

- What is the product sales cycle for mobile phone applications ?
- Is it the same for middleware ?
- Does it ride on the mobile phone sales cycle ?
- Is it the same between consumer and enterprise space ?
- Are there any data ?
- Is this market completely different to anything else ?

Be careful though

- Not much market*ing* data to look through
- The market is happening… happening… happening
- Therefore **you cannot size a non existing market** (at least not that easily

Surely though, there must be some information to use

Sounds familiar ?

Today, for most ISVs the following apply :

- They don’t have their software pre-loaded on handsets
- Their software is not sold by retailers
- Electronic software distribution is the norm
- They offer ‘’try & buy’’
- They sell on-line from portals or their own sites

….lost ?

Profound statement 2

‘’We’re not alone, there are others ‘’

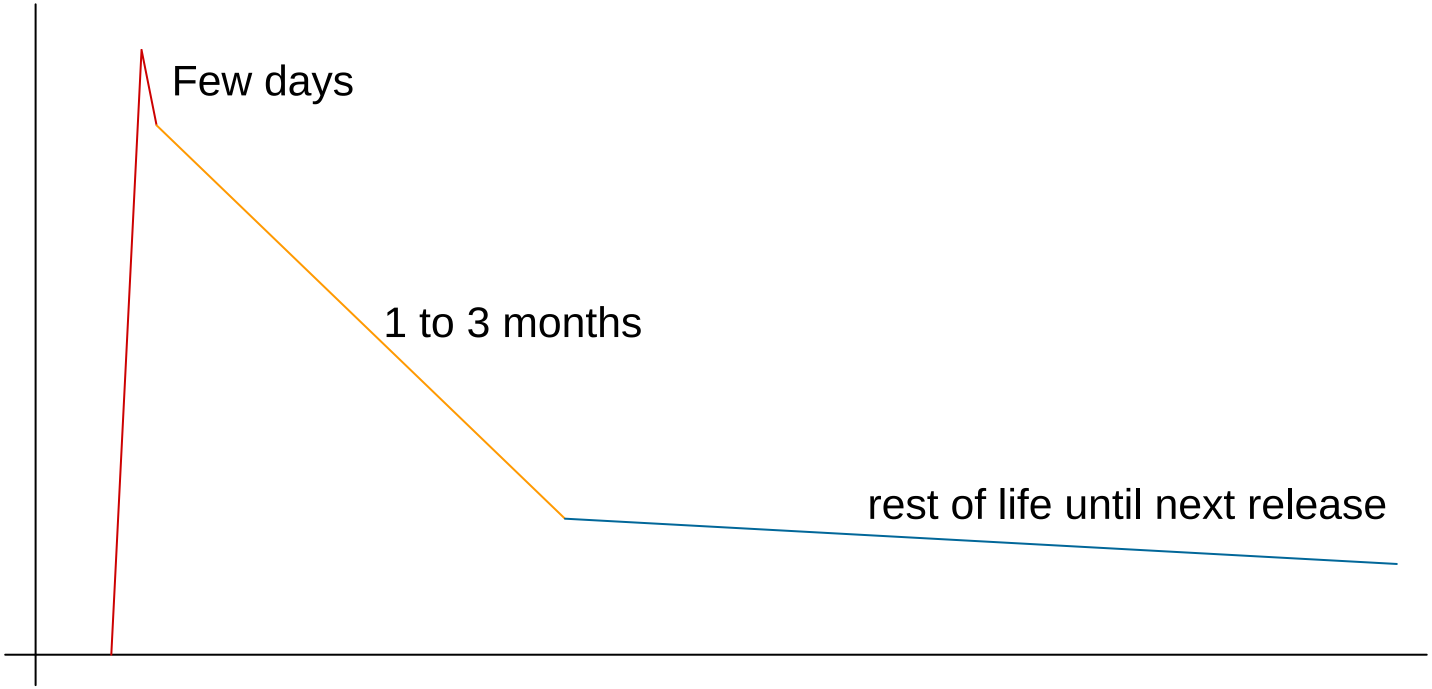
☺

There is a high correlation of the ISV mobile applications market with the PC/Mac shareware market

We can look there for problems and solutions that have been going on for a very long time. There is a lot of information and infrastructure, to learn from.

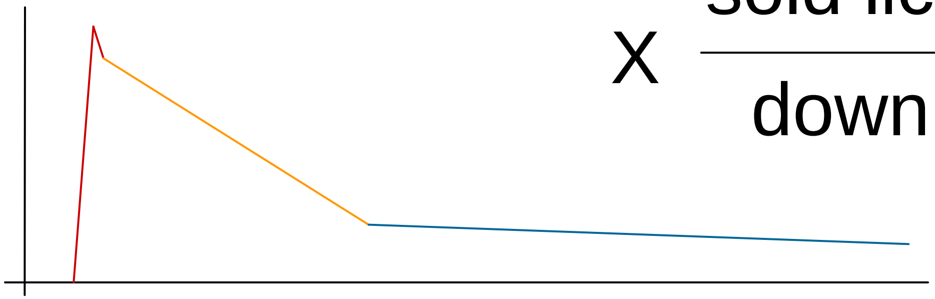
So lets do that….

Sales cycle for electronically distributed applications



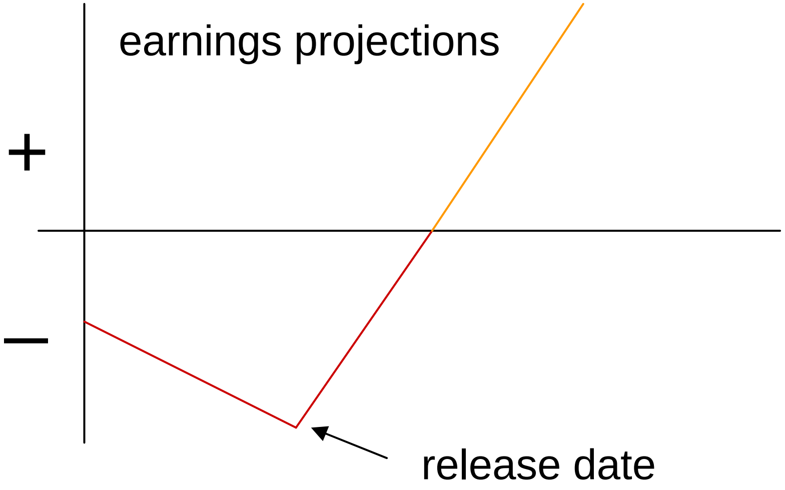
Graph is not to scale, periods may vary for different apps while they vary a lot for games

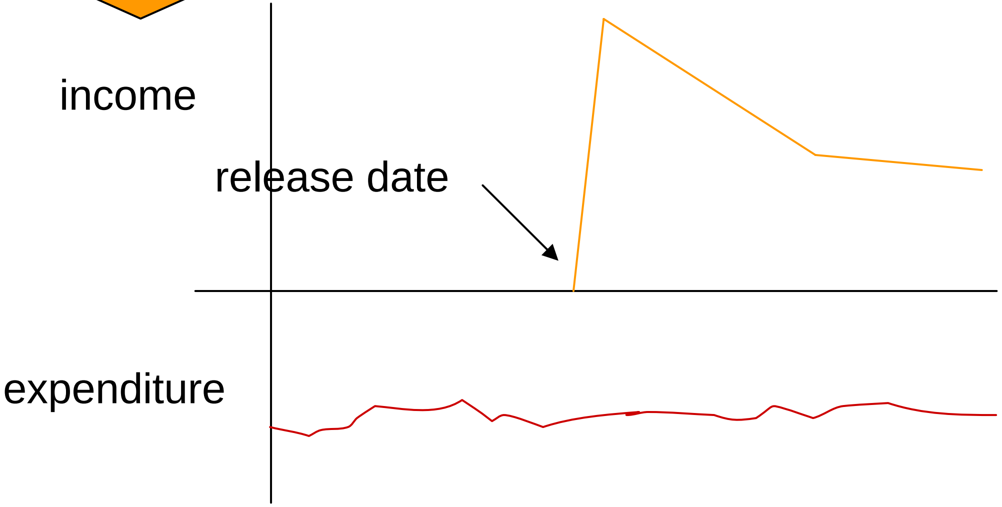
Budgeting



You may want to vary this during the cycle

$$\times \frac{\text{sold licenses}}{\text{downloads}} \times \text{price per license}$$





Things to keep in mind

- Peaks happen around major *and* minor releases.
- During the peak period you will realise most of your revenue, not during the ‘’steady state’’.
- *Release often in order to have more peaks*.
- Your software development efficiency will determine your business sales and vice versa (‘Lean Software Development: An Agile Toolkit for Software Development Managers’,M. Poppendieck & T. Poppendieck)
- You’ll need to produce high quality at low cost or you will be out of the game (Toyota Production System: Beyond Large-Scale Production, Taiichi Ohno)

Things to keep in mind - *be* available

- Your customers buy anytime 24-7-365
- You better be always available to accept the high traffic and payments during the peaks, or you may go out of business fast.

More things to keep in mind

- First make it easy for customers to download ***and*** install, then make it easy to pay.
- The ‘Try & Buy’ trial period will introduce a hysterisis in the downloads to sales conversion.
- If you have defects, or you make your software license or registration process hard, you will be swamped with support requests.

So why do these peaks happen ?

In the shareware business such peaks happen after a release because of the **exposure and promotion** of new software through online (and other) **media**.

- On line reviews
- Press releases
- Blogs, newsgroups and fora
- On line catalogues - ‘What’s new’ lists
- On line advertisements
- Inclusion in magazine CDs
- Reviews in magazines

Electronic Software Distribution Channels

So if this is what happens in the shareware business, do you need portals and catalogues to sell your wares ?

There are two kinds of such portals,

✓the ones that you sell through

✓the ones that you just promote your apps from.

It seems that, as opposed to the personal computer shareware business, in our industry we have so far, mostly portals of the first kind.

An ESD Portal checklist

There will be good and bad portals. Choose *very* wisely

- Do they give you fair contract T&Cs ? Are you in control of price, channel, promotion and brand ?
- What sort of promotion service do they offer ?
- What is the availability of their store ?
- Is their store integrated into your preferred licensing scheme ?
- Who handles the registration support ?
- Do they offer good and fine grained data as to who buys what and from where, how, when etc ?
- At what cost do they offer these services ?
- Some (operator?) portals seem unavoidable, e.g. Java games
- Could *you* do it instead ?

What about DIY ?

- You need to do all the promotion and ad campaigns

  Budget time and money for on line ads, press releases, catalogues, community site promotions, site reviews, banner ads etc

  Spending on online advertising in the UK is estimated to top £1bn. While in the US in the first half of this year spending was around $5.8 bn (according to Interactive Advertising Bureau (IAB) and PriceWaterhouseCoopers (PwC))

- You need to build and maintain the store

  Approx 23% of all on-line shoppers whose purchase fails may never purchase from that site again (''Winning the online consumer''. The Boston Consulting Group, March 2000 study)

Your Store

- Needs to be available …remember the sales cycle
- Needs to cater for the license registration (anyway?)
- Needs to integrate with payment services (multi-currency ?)
- Needs to cater for VAT globally, for both consumers and business users (according to UK/EU regulations at least)
- Needs to record and store invoices for up to 7yrs (in the UK at least) …backups are important for IRS as well :-)
- Needs a payment and customer support channel (not the same as technical support)
- Cater for the Distance Selling Regulations, Data Protection Act (depends on your gov's laws etc)

License registration and locking

- Do you need it ?
- ***Don’t make it hard for people to give you their money***.
- Don’t upset them after they have done so !!!
- Many people who do pay, have registration problems with license keys/tokens.
- Anything you do, better be well integrated with ‘’your’’ store.
- Customers demand good service after the sale

Other routes to market to consumers

- Shrink-wrapped products
- On memory cards through retailers
- Through operator deals
- Through manufacturer deals
- Through retailer deals that offer pick-n-mix services
- Using some purchase agents on the phones (offered by some application portals)

Marketing to business

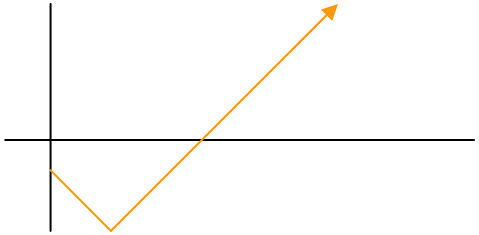
- As components to other developers
- Middleware for enterprises
- Re-branding apps for operators
- Bespoke development, using your pre-existing components
- Symbian Affiliate programme

Today online , tomorrow in the box

Remember that the ISV apps of today become the apps that get included with the phones tomorrow either by operators, manufacturers or retailers.

The chances are that we will not get a Visicalc but rather a cocktail of apps, that makes consumers want to buy a particular phone.

I hope your innovations create many such cocktails and that you're very successful.



Thank

You