Symbian code signing

or – how to implement a mobile code signing scheme

Overview - theory

- History of OS
- The Symbian Foundation
- Future outlook

Overview - coding

- Development options
- Native applications in C++

- Selling an application

- Where to go for help

Overview – security I

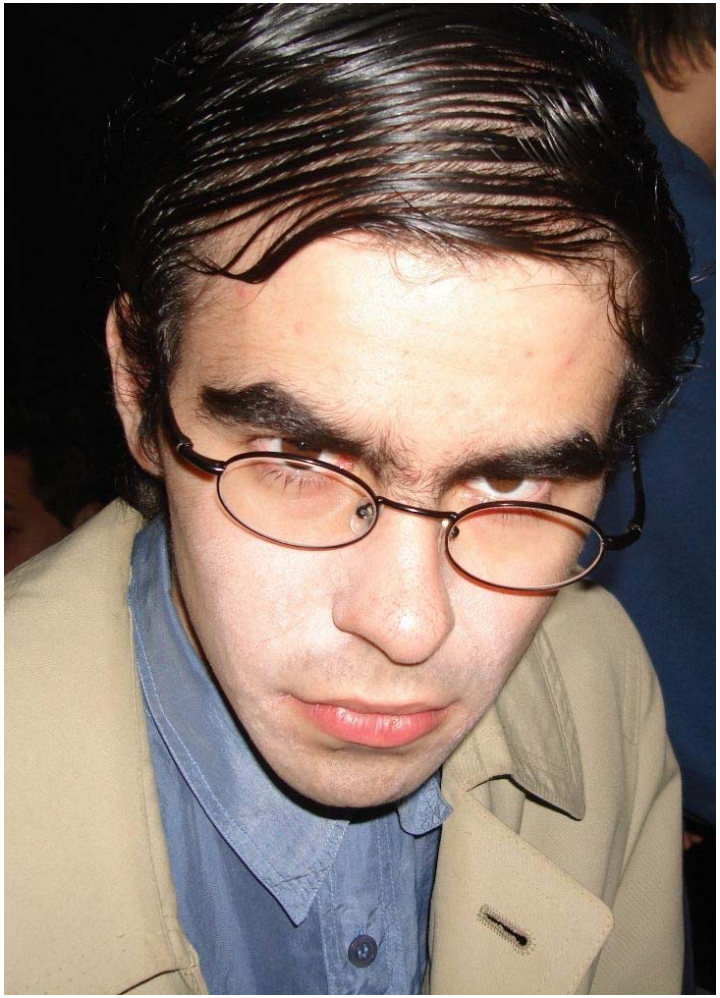
- Why handset-based security
- Stakeholders in mobile security

- Architectural goals of the Platform Security Model
- Pillars of the Platform Security Model

Overview – security II

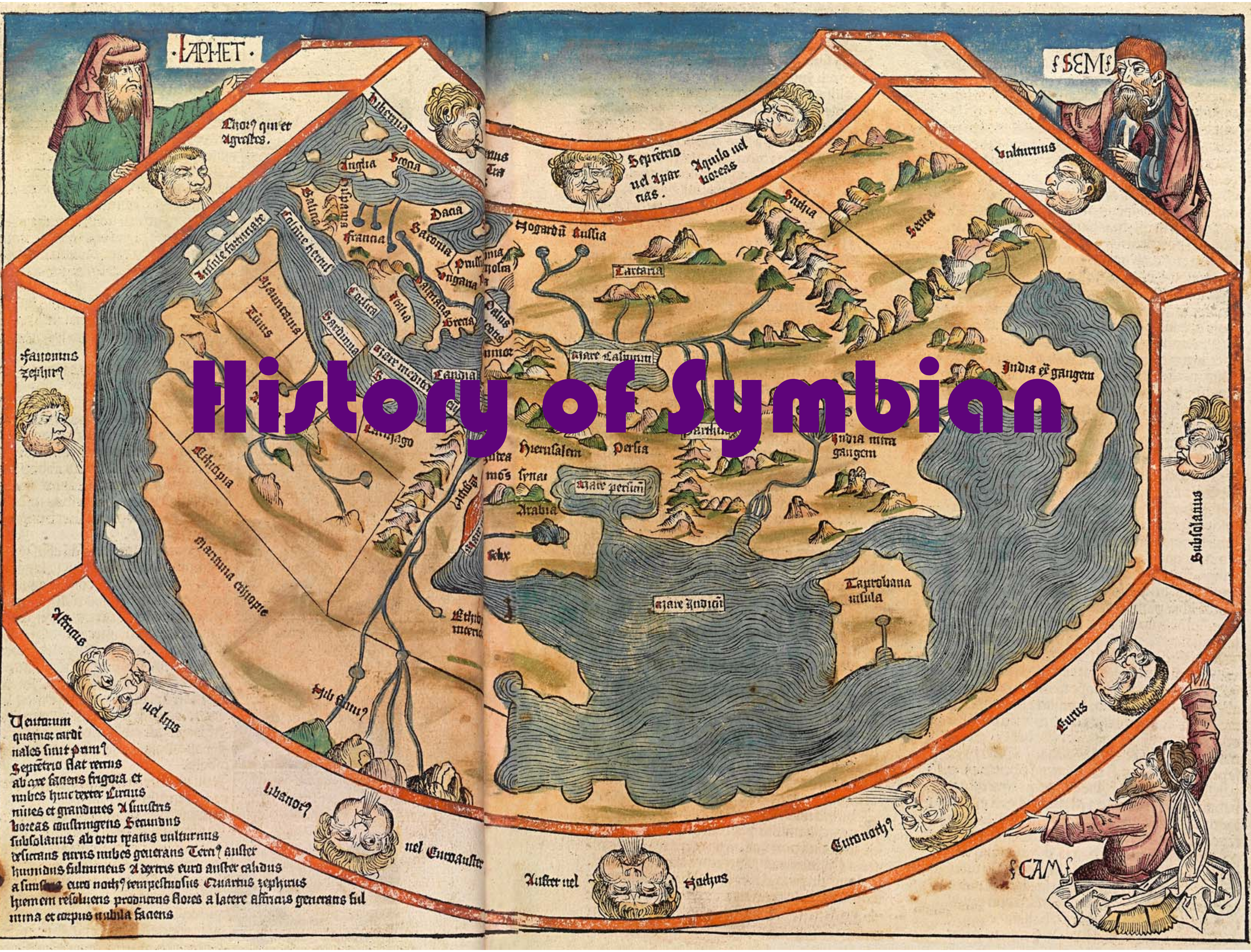
- The Native Software Installer
- Signing games

About /me



- Tam HANNA
  - CEO, Tamoggemon Ltd.
  - Runs web sites about mobile computing

History of Symbian



EPOC

- Developed by Psion
  - Series5
  - Revo

- "Thrown out" to Symbian

Series60

- First versions
  - Introduced on 7650
- S60v2 still common
  - Nokia N70

S60v3

- Renamed due to virus problems
- Introduces mandatory signing
  - Binary break
- Three feature packs
  - Downward compatible

S60v5

- S60v3 + touch
  - Lives along v3
- Very basic GUI
- Partially downward compatible
  - Apps run, but cant be controlled due to lack of buttons

The Symbian Foundation

Who owns it

- Non-profit organization
  - Governed by boards
  - Members picked on "amount of merit"
- Founded by various cell phone companies

- Every organization can become a member
  - "Smallish" software houses can also join

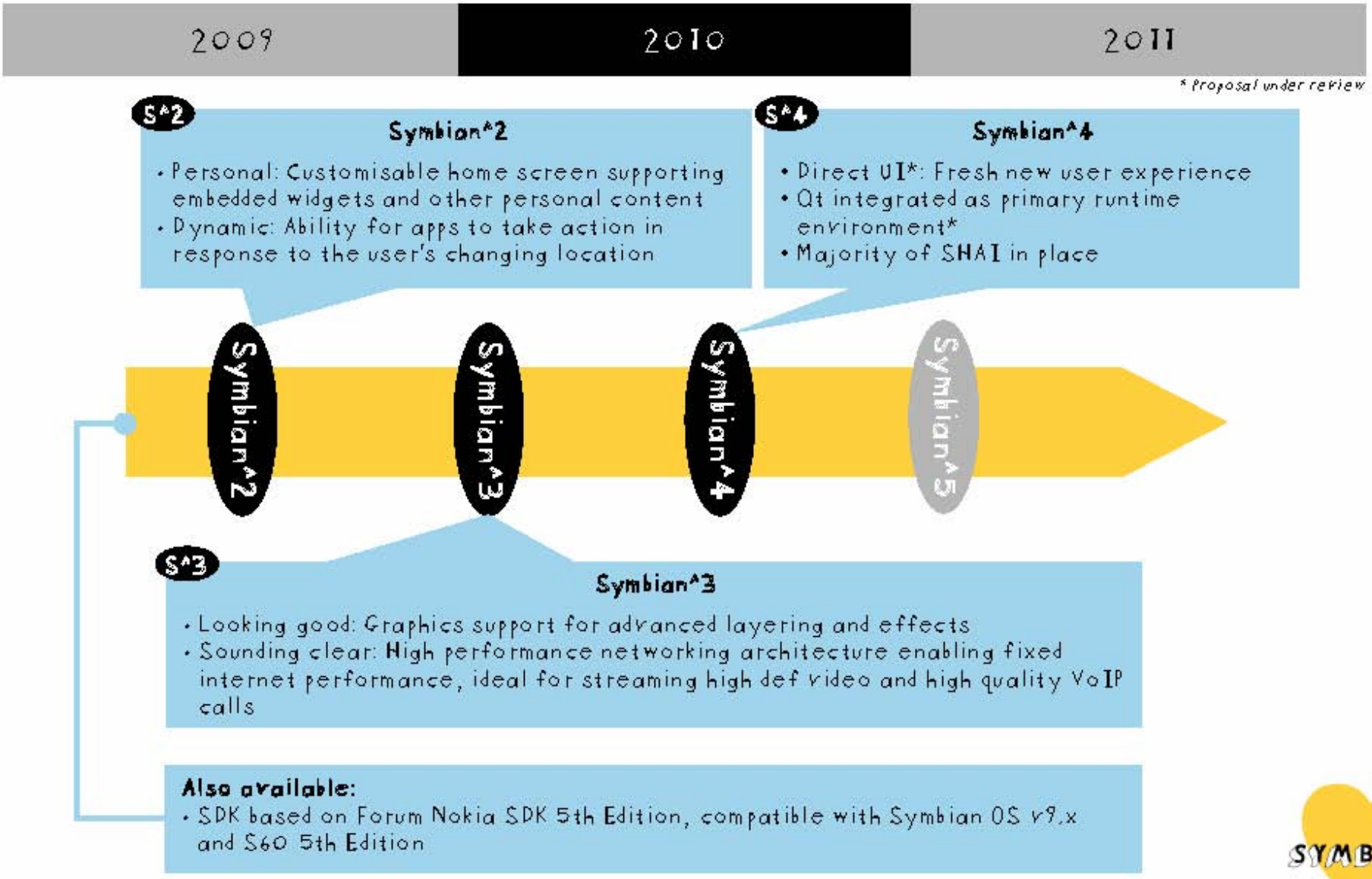
licensing

- Code is currently being moved to EPL
  - Expected done in mid-2010
  - Parts out earlier
- Currently, members can access all code
  - Symbian Public License


FUTURE
66
future outlook

The Symbian Foundation Platform Plan

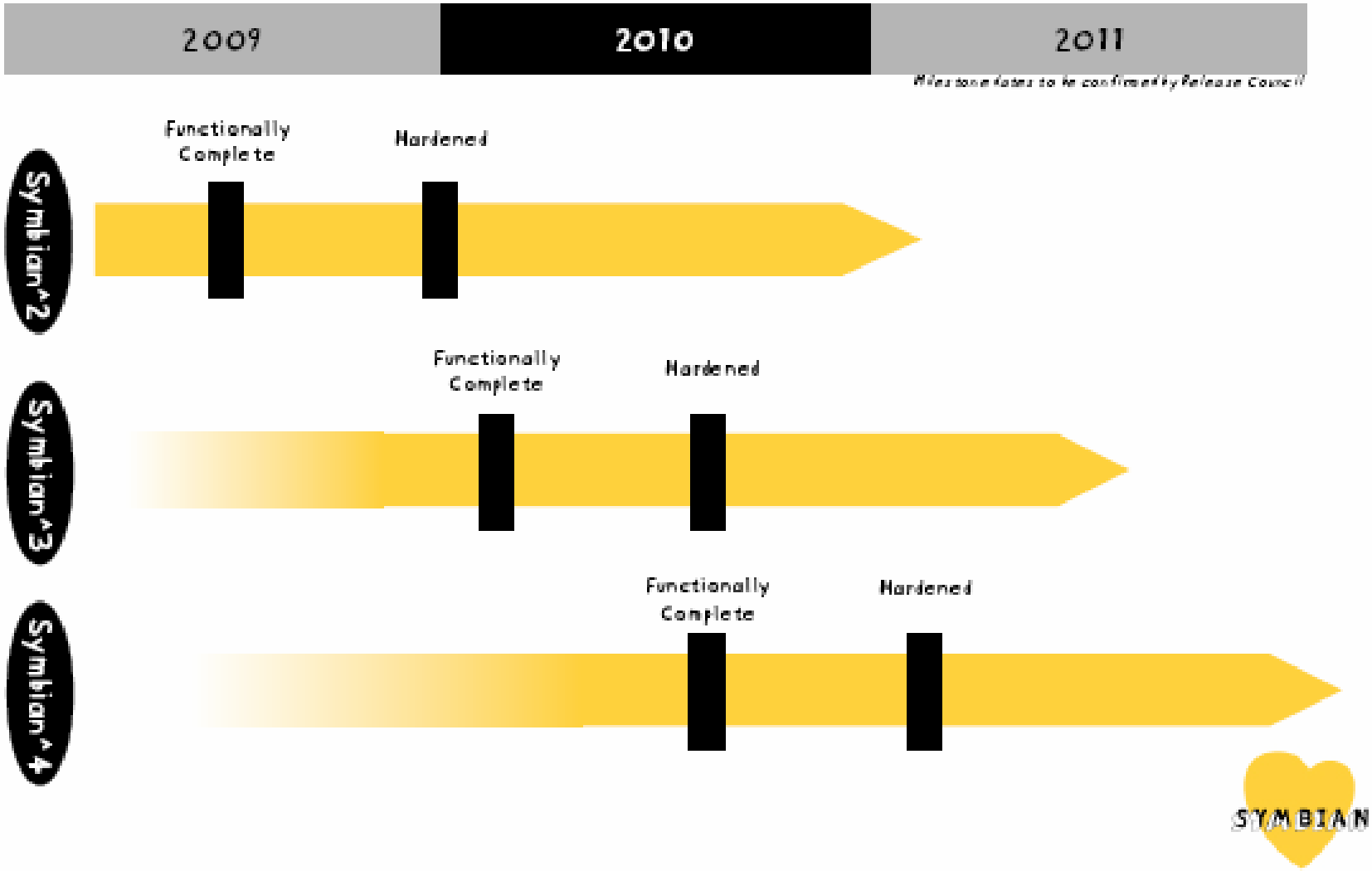
Open for contribution



SYMBIAN

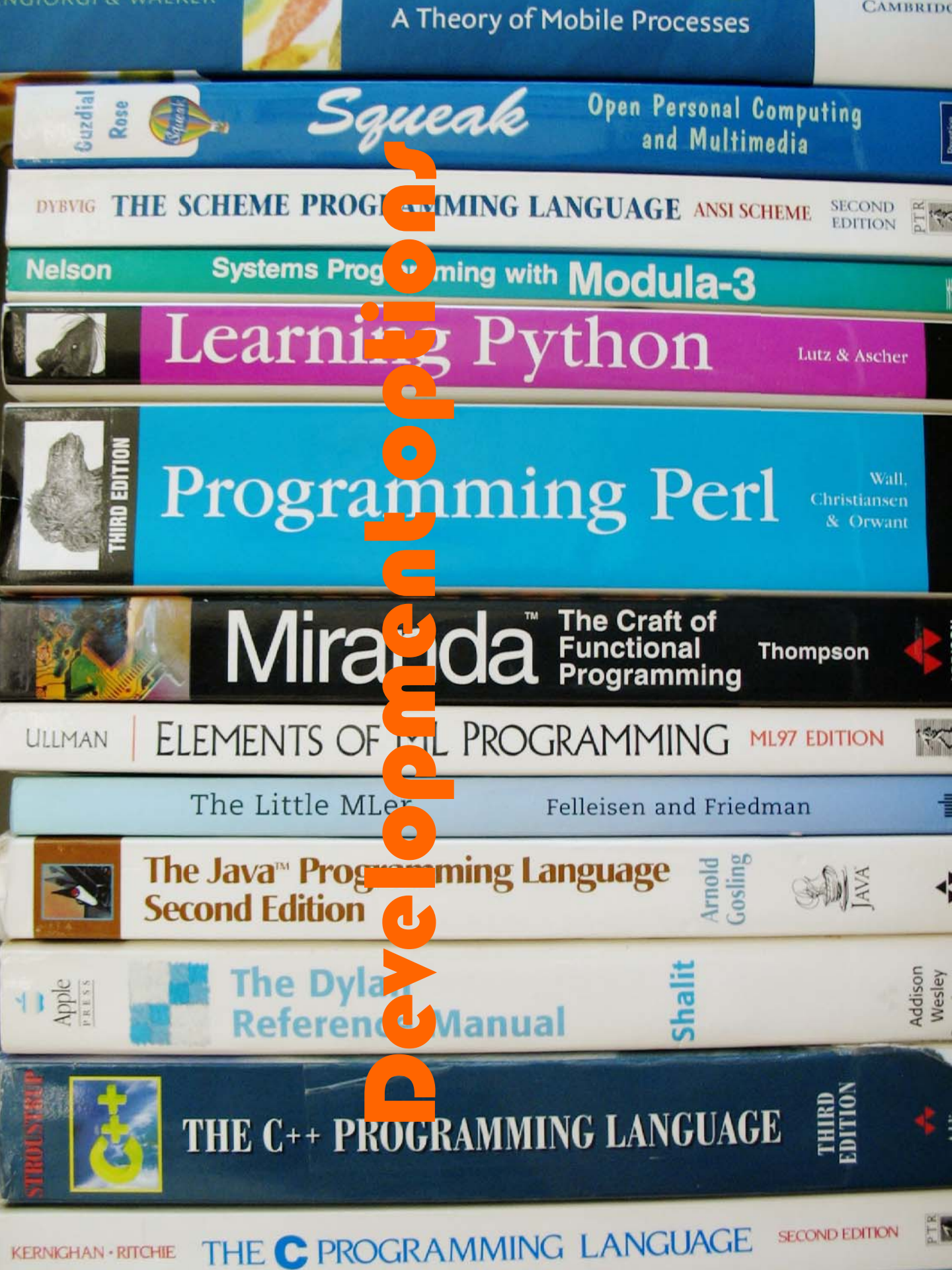
Symbian Foundation Platform Plan

Open for contribution



What to expect

- Will switch to QT
  - Known to open-source heads
  - Native development will remain possible
- Symbian^4 – new UI stack
  - LARGE binary break


A Theory of Mobile Processes
Guzdial
Rose
Squeak
Open Personal Computing and Multimedia
DYBVIG
THE SCHEME PROGRAMMING LANGUAGE
ANSI SCHEME
SECOND EDITION
PTR
Nelson
Systems Programming with Modula-3
Learning Python
Lutz & Ascher
THIRD EDITION
Programming Perl
Wall, Christiansen & Orwant
Miranda™
The Craft of Functional Programming
Thompson
ULLMAN
ELEMENTS OF ML PROGRAMMING
ML97 EDITION
The Little MLer
Felleisen and Friedman
The Java™ Programming Language
Second Edition
Arnold
Gosling
JAVA
Apple PRESS
The Dylan Reference Manual
Shalit
Addison Wesley
STROUSTRUP
C++
THE C++ PROGRAMMING LANGUAGE
THIRD EDITION
KERNIGHAN · RITCHIE
THE C PROGRAMMING LANGUAGE
SECOND EDITION
PTR
Developmentor

C++

- Symbian "fooked" C++
  - "C with objects"
- Somewhat difficult to learn and understand

- But:
  - Apps somewhat easy to sell
  - Full device access

.NET CF

- Net60
  - from third party
- Must be purchased
- Great communality with PPC et al

J2ME

- Oldest run time environment
- Developing for it is somewhat easy

- But:
  - Bad user experience (prompts)
  - Might be dropped from future boxen

WRT / flash / etc

- Various Web 2.0-Technologies available
- Easy to develop for

- But:
  - Selling results is difficult at times
  - Extremely limited functionality


Native applications in
C++

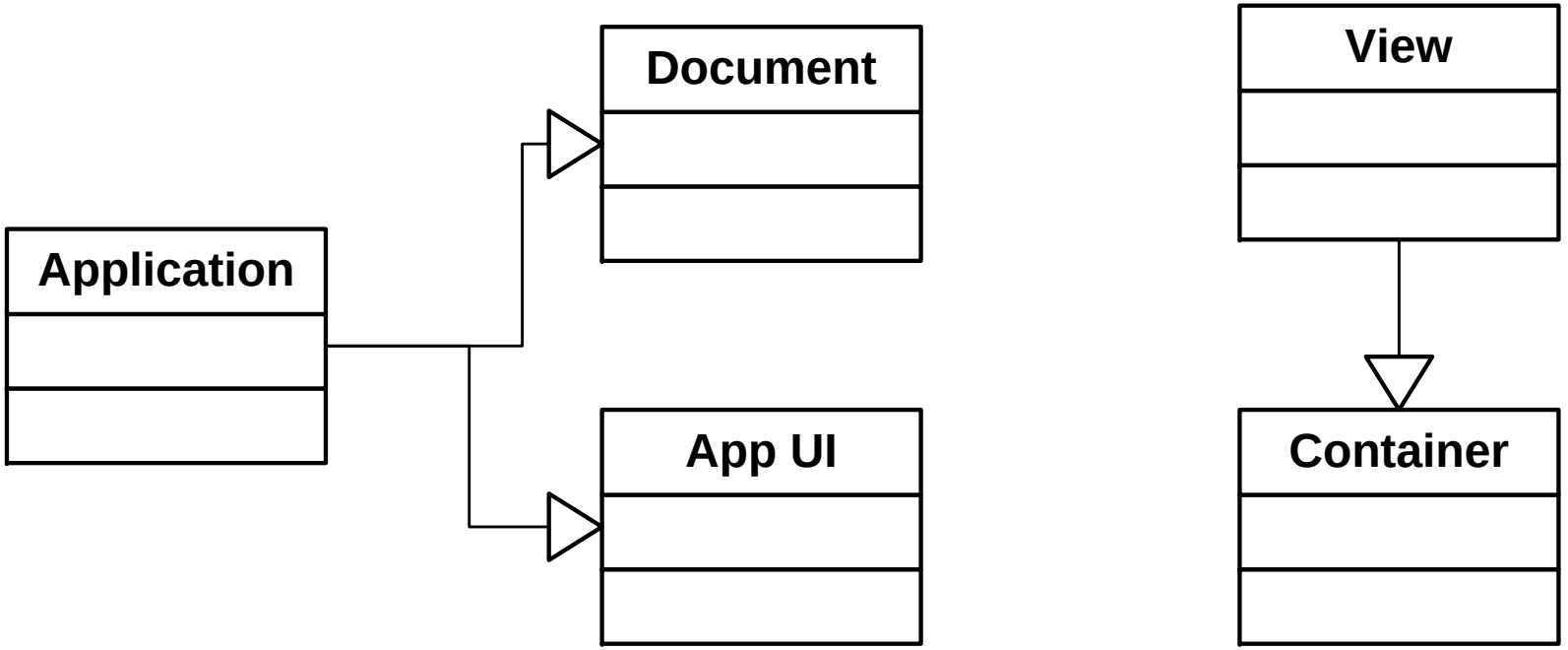
Carbide.c++

- Free, Eclipse-based IDE

- Contains dangerously buggy UI editor
  - COMMIT CODE TO CVS FREQUENTLY!!!

Basic app structure

- Five classes:



- Auto-generated

Application / Document

- Do not touch
- Let Carbide generate them, and ignore
- Development mantra: if it works, its ok
  - FOOK the rules

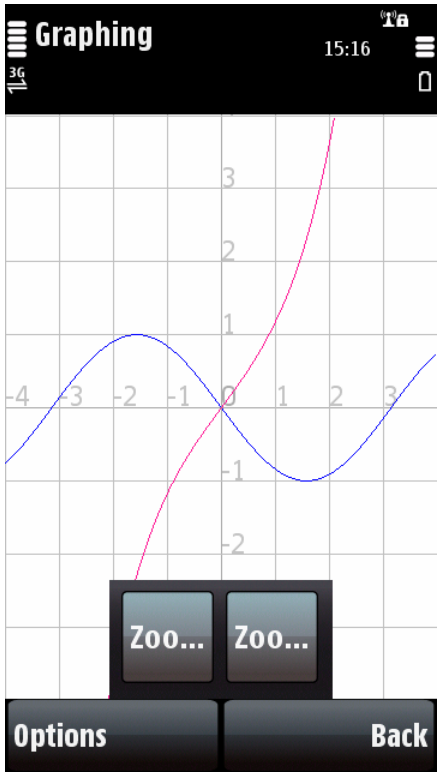
AppUi

- Use as "static repository"
  - Stores global data
- Can be accessed easily from forms

- CLocaNoteAppUi* temp = (**static_cast**<CLocaNoteAppUi*>((CCoeEnv::**Static**()) ->**AppUi**()))

View / Container

- Contain logic for "forms"
- One view, one container per form



Two-phase initialization

- Handsets always low on memory
- "Clean up stack" catches object references
  - Stack unwinding, clean-up
- Can generally be ignored

```
/**
 * First phase of Symbian two-phase construction. Should not contain any
 * code that could leave.
 */
CTouchCalcContainerView::CTouchCalcContainerView()
{
// [[[ begin generated region: do not modify [Generated Contents]
iTouchCalcContainer = NULL;
// ]]] end generated region [Generated Contents]

}

/**
 * The view's destructor removes the container from the control
 * stack and destroys it.
 */
CTouchCalcContainerView::~CTouchCalcContainerView()
{
// [[[ begin generated region: do not modify [Generated Contents]
delete iTouchCalcContainer;
iTouchCalcContainer = NULL;
// ]]] end generated region [Generated Contents]
```

- }

```
/**
 * Symbian two-phase constructor.
 * This creates an instance then calls the second-phase constructor
 * without leaving the instance on the cleanup stack.
 * @return new instance of CTouchCalcContainerView
 */
CTouchCalcContainerView* CTouchCalcContainerView::NewL()
{
CTouchCalcContainerView* self = CTouchCalcContainerView::NewLC();
CleanupStack::Pop( self );
return self;
}

/**
 * Symbian two-phase constructor.
 * This creates an instance, pushes it on the cleanup stack,
 * then calls the second-phase constructor.
 * @return new instance of CTouchCalcContainerView
 */
CTouchCalcContainerView* CTouchCalcContainerView::NewLC()
{
CTouchCalcContainerView* self = new ( ELeave ) CTouchCalcContainerView();
CleanupStack::PushL( self );
self->ConstructL();
return self;
}
```

```
/**
 * Second-phase constructor for view.
 * Initialize contents from resource.
 */
void CTouchCalcContainerView::ConstructL()
{
// [[[ begin generated region: do not modify [Generated Code]
BaseConstructL( R_TOUCH_CALC_CONTAINER_TOUCH_CALC_CONTAINER_VIEW );

// ]]] end generated region [Generated Code]

// add your own initialization code here

}
```

Active objects

- Primitive form of multi-tasking
- An AO runs “in the background”

- Needed for some API calls

```
void CSaveImageAO::SaveAsL(CFbsBitmap* aBitmap, TFileName& aFileName, TUid aImageType,
    TUid aImageSubType)
  {
    iImageEncoder->Convert( &iStatus, *aBitmap);
    SetActive();
    }
  }
void CSaveImageAO::DoCancel()
  {
  iImageEncoder->Cancel();
  }
void CSaveImageAO::RunL()
  {
  iObserver->ImageSaved(iStatus.Int());

  }
TInt CSaveImageAO::RunError(TInt aError)
  {
  iObserver->ImageSaved(aError);
  return KErrNone;
  }
```

-

Testing

- Simulator
  - Very slow
  - With HookLogger => mem leak buster
- Real device (might need DevCert)
  - Faster
  - Works via USB
  - Install Nokia PC Suite, TRK


Selling applications

Signing

- Express signed – 20$
  - Doesn't permit all capabilities
  - Usually no checks done
- Certified signed – 200$
  - Permits most capabilities
  - Thorough checks (!!!)

Signing - II

- Certificate needed
  - Obtained from TrustCenter Germany
- Requires registered company
  - Recommendation: UK Limited
  - www.go-ahead.de

Ovi Store

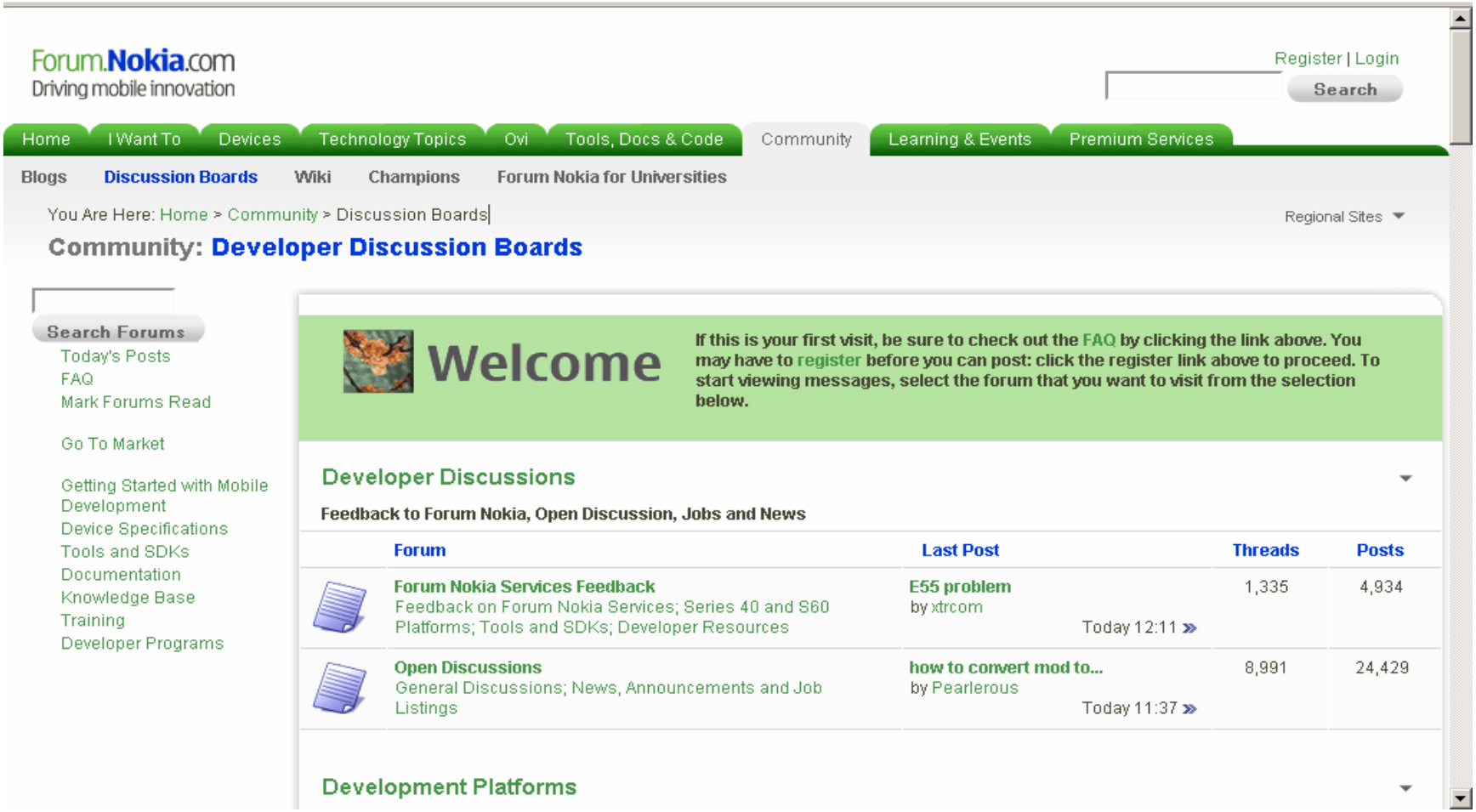
- Nokia's sale channel
- 70% of gross
- Somewhat slow ATM

Traditional channels

- Various traditional ESD's
  - Handango
  - MobiHand
- Diminishing importance


Where to go for help

Forum Nokia


Forum.Nokia.com
Driving mobile innovation
Register | Login
Search
Home
I Want To
Devices
Technology Topics
Ovi
Tools, Docs & Code
Community
Learning & Events
Premium Services
Blogs
Discussion Boards
Wiki
Champions
Forum Nokia for Universities
You Are Here: Home > Community > Discussion Boards
Regional Sites
Community: Developer Discussion Boards
Search Forums
Today's Posts
FAQ
Mark Forums Read
Go To Market
Getting Started with Mobile Development
Device Specifications
Tools and SDKs
Documentation
Knowledge Base
Training
Developer Programs
Welcome
If this is your first visit, be sure to check out the FAQ by clicking the link above. You may have to register before you can post: click the register link above to proceed. To start viewing messages, select the forum that you want to visit from the selection below.
Developer Discussions
Feedback to Forum Nokia, Open Discussion, Jobs and News
Forum
Last Post
Threads
Posts
Forum Nokia Services Feedback
Feedback on Forum Nokia Services; Series 40 and S60 Platforms; Tools and SDKs; Developer Resources
E55 problem
by xtrcom
Today 12:11
1,335
4,934
Open Discussions
General Discussions; News, Announcements and Job Listings
how to convert mod to...
by Pearlerous
Today 11:37
8,991
24,429
Development Platforms

Forum Nokia - II

- Free to register and use
  - But: bad search engine
- Most questions get answered quickly
  - Nokia employees around

- http://discussion.forum.nokia.com/forum/

Nokia Wiki

- Nokia-operated Wiki
- Contains valuable tutorials

Symbian Foundation

- Has developer help services of its own
- Less developed than Nokia's

- =>Stick to Nokia until told otherwise

Why handset-based security

Different user perceptions

- Mobile phones are always on the user
  - More personal
- User feels that unit "is safe"
  - No large-scale outbreaks so far
  - User is unwilling to accept implications of AV software

Carriers can't do it alone

- GSM / CDMA
  - Can be protected
- Bluetooth
  - Can't really be protected by the carrier
- WiFi
  - Don't even ask

Users are stupid

- Cabir displayed THREE warning alerts
  - Perimeter security is not enough
- Users choose dancing pigs over security
  - Ed Felton


Stakeholders

Carriers

- Don't want to invest $$$
  - Don't burden us with investments / infrastructure
- Don't want to deal with unhappy users
  - Keep them happy
- Gain revenue if users buy more apps
  - Feeling safe == more app sales

Developers

- Want easy access to full OS features
  - Will move to other platform if not given
- Want simple development process

- Gain revenue if users buy more apps
  - Feeling safe == more app sales
- Less risk if bugs occur
  - Can't access dangerous stuff

OS vendor

- Doesn't want large virus outbreaks
  - Bad PR
- Doesn't want to piss off developers
- Doesn't want to piss off users

- Doesn't care much about power users

User

- Doesn't want to be bugged
  - J2ME, anyone?
- Doesn't want battery drain, etc
  - Caused by AV activity
- Wants cheap apps
- Wants data to be safe

User (power user)

- Wants full access to the system
- Wants powerful apps
- (Gets f##ed most of the time)

Architectural goals of PSM

Ensure Understandability

- Users are the weakest point of secure systems
- Users don't understand technology
- DON'T offload decisions to them
  - No IE-like prompts

Support open phones

- Successful app market == Successful OS
- Minimize impact on legitimate developers
  - But keep jerks out

Protect the network

- Carriers want their networks to be safe
- Software may NEVER damage the network
  - More carriers will use the OS
  - Larger market

Be light-weight

- Preserve CPU cycles

- Don't do unnecessary checks
  - Less than 40% of API is "managed"
  - Access rights are computed at start-up of the app

Provide a basis for trust

- Make users trust their phones
  - More app sales
  - More OS sales
- Everybody benefits


The pillars of PSM

The Process - I

- Mobile phone users are usually "authorized"
  - No multi-user phones
  - PIN Authentication
- User-based rights management doesn't make sense

The Process - II

- Processes are the smallest sensible unit
- **The Process is the Unit of Trust**

- 1 process = 1 app

- Processes are divided into tiers

Software installer
System services
Kernel
F32
nsigned stuff
tuff
gr

The capability

- **A capability is a token which must be presented to gain access to a privileged service**
- Come in three classes
  - TCB
  - System
  - User

The capability - II

- TCB Capabilities: TCB
- Granted to TCB processes only
- Lets them do things nobody else can

The capability - III

- System Capabilities
  - Not meaningful to user
  - Granted by a signing house
- User Capabilities
  - "Not really dangerous"
  - Granted by user (like J2ME)

Data caging

- Access to some folders is restricted
- Provides “secure storage”

- But: MMC/SD readers

Data caging - II

Path	Read	Write
/sys	AllFiles	TCB
/resource	-	TCB
/private/mySID	-	-
/private/notMe	AllFiles	AllFiles
/other	-	-

Capabilities

An overview

Capability eekers

- 1 capability? 2000 capabilities?
- Granularity must be set up reasonable
- Symbian has 20 capabilities

TCB

- Write to executables
- Write to read-only rsrc files

- Not usually given out – MANUFACTURER

Allfiles

- Read access to the entire FS
- Not usually given out - MANUFACTURER
  - Caused the death of third-party file managers

DRM

- Access to DRM-protected content
- Not usually given out – MANUFACTURER

CommDD

- Direct access to Wifi, etc hardware / drivers

DiskAdmin

- Mount, unmount file systems

MultimediaDD

- Direct access to camera, etc drivers
- “Priority multimedia access”

NetworkControl

- Control network protocols

- E.G. Close all TCP/IP links
- Set network defaults

PowerMgmt

- Kill processes
- Turn off box
- Disable peripherals

ProtServ

- Register protected server
  - Name with ! At the beginning

ReadDeviceData

- Read data like:
  - PIN
  - List of installed apps

SurroundingsDD

- GPS / biometrics driver access

SwEvent

- Handle and dispatch key, pointer events GLOBALLY

TrustedUI

- Display trusted dialogs

WriteDeviceData

- Change things like:
  - Time zone
  - Device lock
  - System Time

localServices

- Access to BT, IR, ...
  - May NOT cost user $$$
- USER-granted

location

- Access to GPS coordinates
- USER-granted

NetworkServices

- Access to GSM/EVDO
  - Might cost user $$$
- USER-granted

ReadUserData

- Contacts
- Messages
- Appointments

- USER-granted

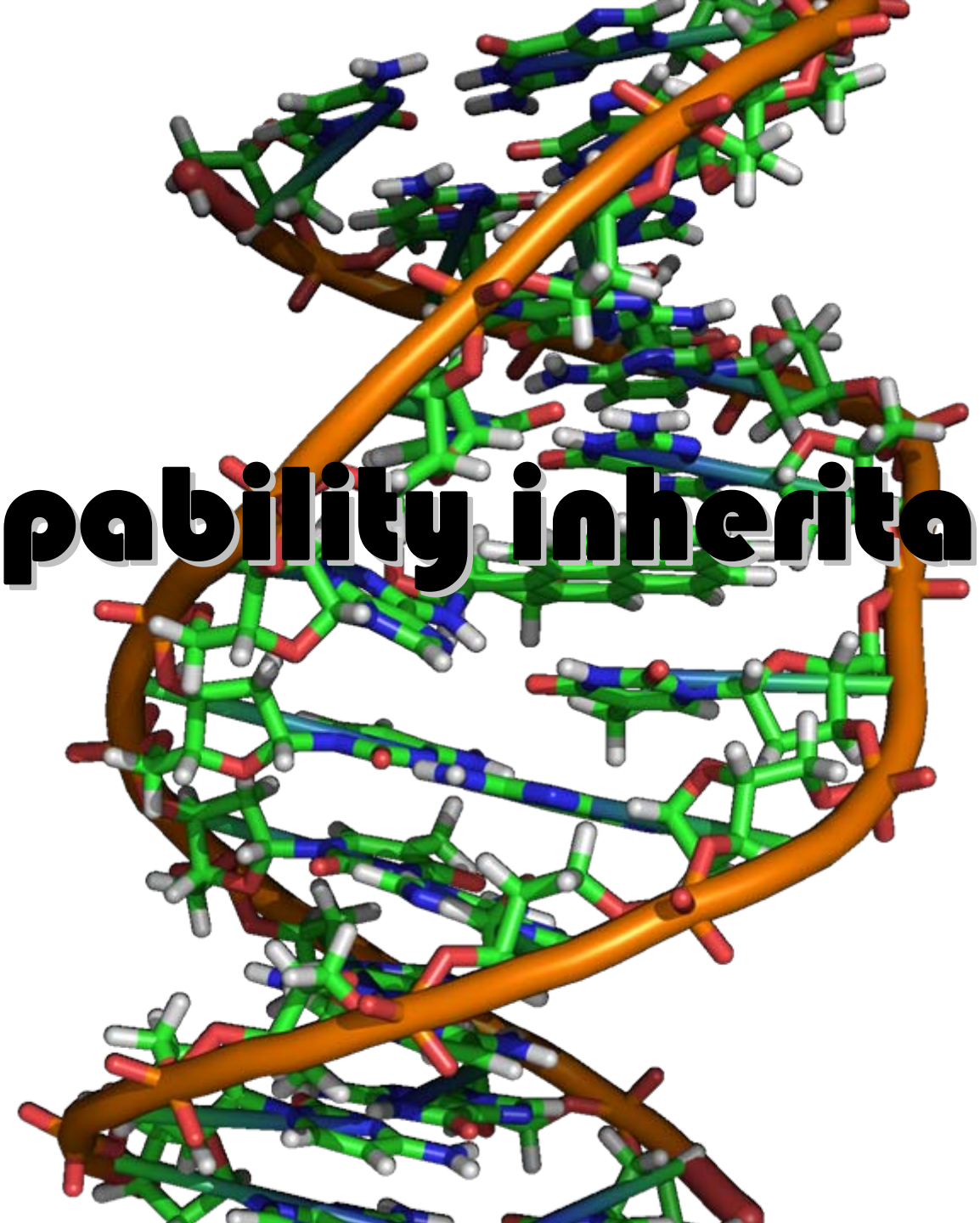
UserEnvironment

- Access to recording, etc at API level
- USER-granted

WriteUserData

- Write access to “user data”
- Depends on device
- USER-granted

Capability inheritance

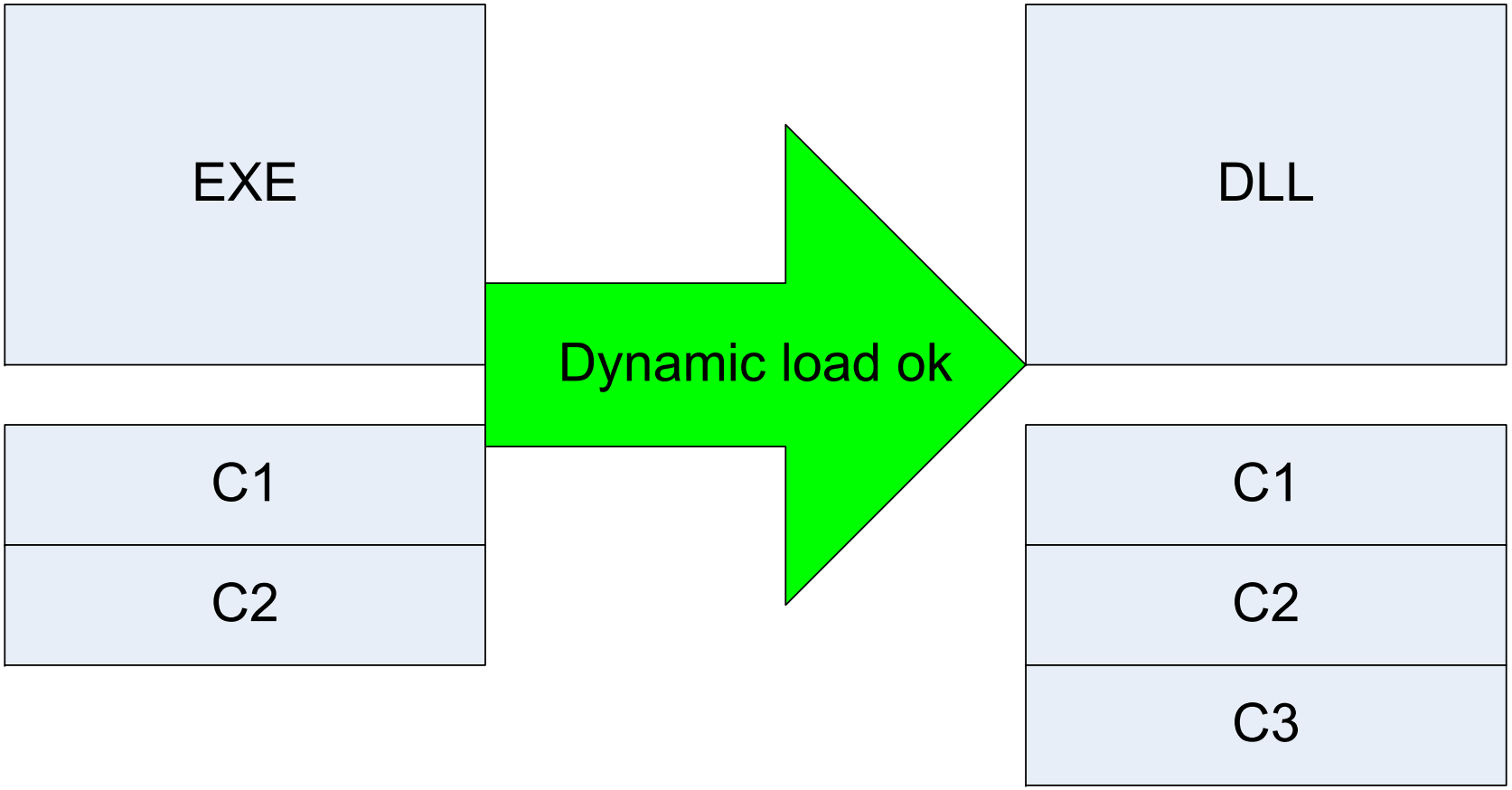


Who cares / Why care?

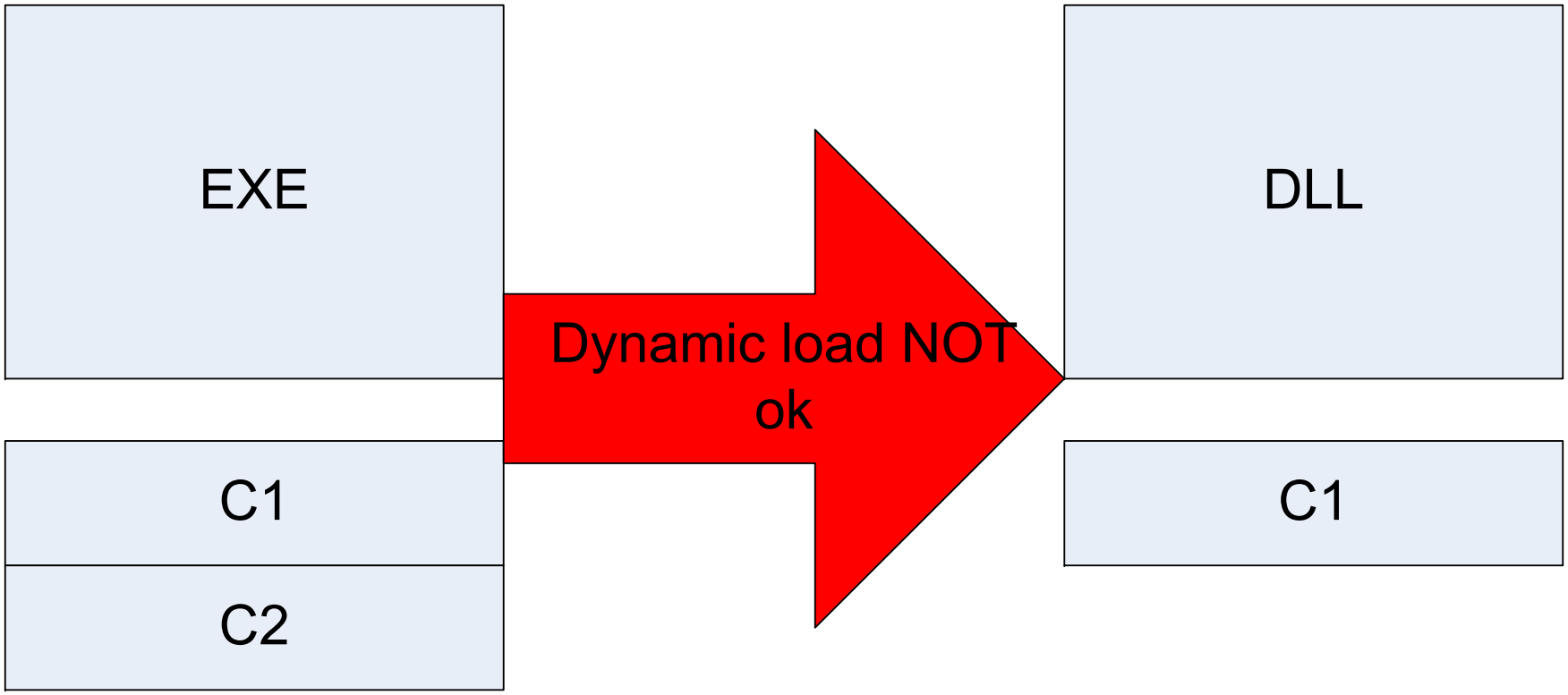
- My capabilities are limited?
- I can spawn a process from another DLL
- It has more privileges than I do
- Uh-Oh!

Direct loading of Dll

- Don't allow DLLs to import malicious code



Dynamic loading - II



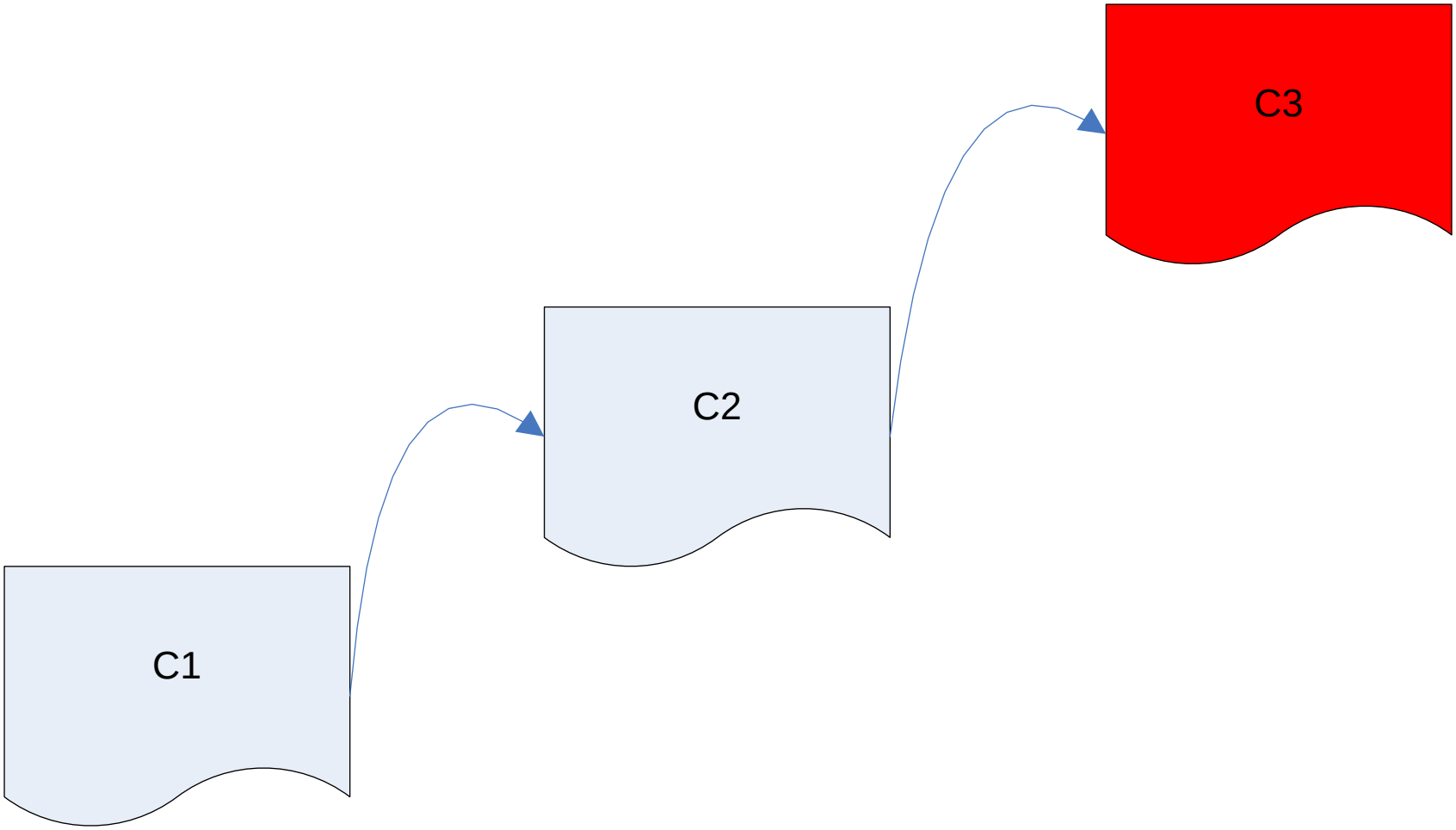
The native software installer

What does he do?

- Acts as gatekeeper between apps and system

- Performs signature check
- Performs capability check
- Handles file moving

The signature chain

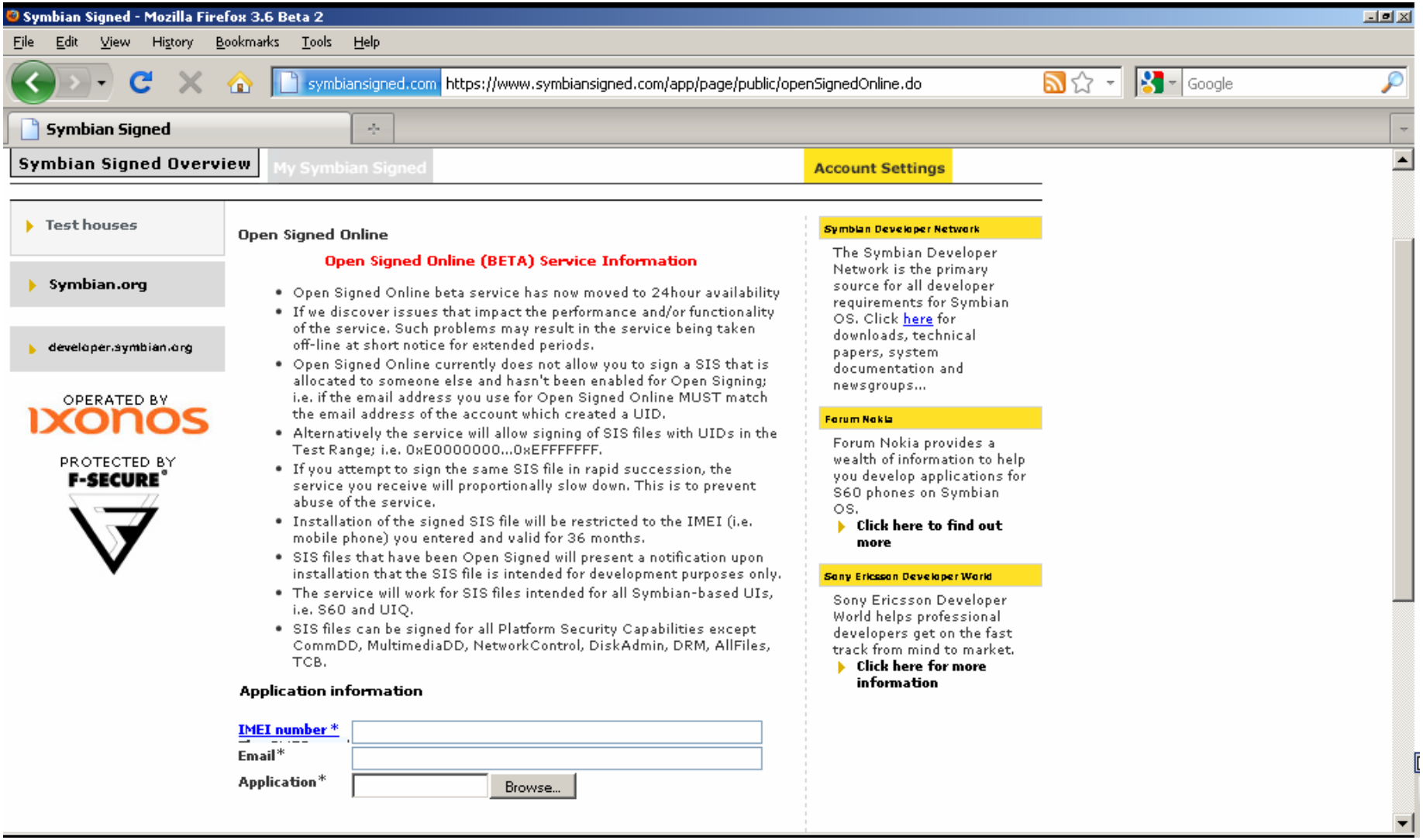


But: who manages the trust?

- Somebody has access to the "root" certs
- This somebody: signing house


Signing games

Open Signed Online



The publisher ID

- Binds ID to company
  - Given out by Trust Center
- Elementary “trust token”
- Needed for successive signings

Developer certificate

- Limited by IMEI
- Allows self-signing for testing purposes

Express signed

- Gives “full signing”
  - But not all capabilities
- Not every app is checked
- Lower cost (approx 20$)

Certified signed

- Thorough checks
- Gives almost every capability
- High cost (at least 200$)

The end

Questions?

Answers at tamhan@tamoggemon.com

images by Julius Kusuma, Cimexus of Canberra, adactio, 3dh3m, Snowmanradio, zephyris, Paul Goyette, Agnostic Preachers Kid