

Position



- Browser fragmentation
- WebKit derivative
- Leveraging WebKit for CE

Browser Fragmentation



- Wide range of offering in the Browser's space for Desktop
- Differents in Embedded constraints

Browser Fragmentation



- NetFront
- OpenWave
- Opera

Royalty-based, Proprietary

Browser Fragmentation



- Firefox, Fennec: not ready for the game
- WebKit
 - Pleyo has been pushing WebKit for embedded systems since 2006
 - iPhone and Nokia S60 Series proved this choice to be sharp
 - LiMo, Android Adoption

Paving the way to broader WebKit adoption



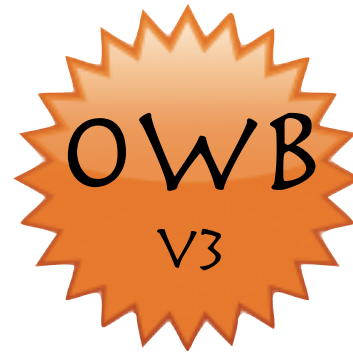
- WebKit for iPhone: specific MobileSafari proprietary code
- WebKit for S60: months of refactoring engineering. Project is dead-born...And dead
- Almost definitively forked from WebKit trunk because of this huge refactory

Paving the way to broader WebKit adoption



- Offer a simple way to port WebKit
- Saves the client from WebKit update concern
- Leverage code reuse from a port to another





OWB



- OWBAL Abstraction Layer: simplify WebKit port
- Extensive and automated testing
- Warrant of a clean WebKit Port
- Respects Device & System limitations

OWB



- Library-dependent instead of Platform-dependent
- Freedom of choice : a feature set associated to libraries should or shouldn't be integrated



OWB Control API

WebKit

Optimizations

AddOn engine

Javascript API

Javascript API

AddOns
Implementation

AddOns
Implementation

WKAL

Private BAL

Graphics

Network

Types

OWBAL

...

Middleware, Libs, OS...



OWB Control API

WebKit

tions

engine

Javascript API

Javascript API

Objects connected
to WebKit

AddOns
Implementation

AddOns
Implementation

WKAL

Private BAL

Graphics

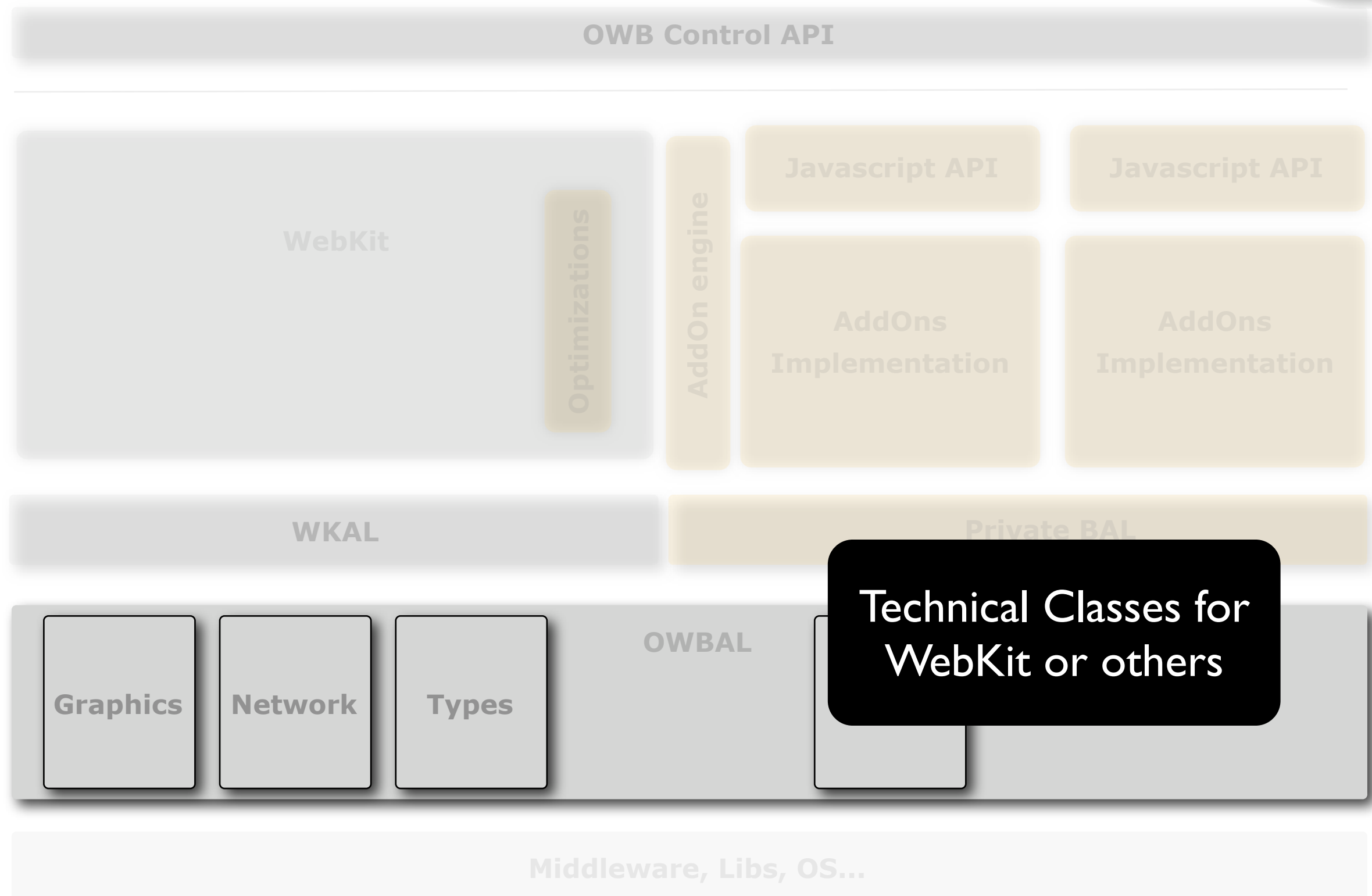
Network

Types

OWBAL

...

Middleware, Libs, OS...



OWB Control API



- Inspired by Win32 WebKit
 - Configuration Management
 - URL Management (Stop, Reload, Back, ...)
 - History Management
- Seamless Integration with foreign apps
 - WebView management
- Platform specific features: Clipboard, Download...

OWB Control API



```
WebCore::BackForwardList* getBackForwardList();
bool goToBackForwardItem(WebCore::HistoryItem* item);
bool canGoBack();
bool canGoBackOrForward(int steps);
bool CanGoForward();
void goBack();
void goBackOrForward(int steps);
void goForward();
void stopLoading();
void open(const char* uri);
void reload();
void loadString(const char* content, const char* content_mime_type,
const char* content_encoding, const char* baseUri);
void loadHtmlString(const char* content, const char* base_uri);
bool searchText(const char* string, bool case_sensitive, bool
forward, bool wrap);
int markTextMatches(const char* string, bool case_sensitive, int
limit);
void SetHighlightTextMatches(bool highlight);
```

**Sample from
WebView**

OWB Control API



```
WebCore::BackForwardList* getBackForwardList();
bool goToBackForwardItem(WebCore::HistoryItem* item);
bool canGoBack();
bool canGoBackOrForward(int steps);
bool CanGoForward();
void goBack();
void goBackOrForward(int steps);
void goForward();
void stopLoading();
void open(const char* uri);
void reload();
void loadString(const char* content, const char* content_mime_type,
const char* content_encoding, const char* baseUri);
void loadHtmlString(const char* content, const char* base_uri);
bool searchText(const char* string, bool case_sensitive, bool
forward, bool wrap);
int markTextMatches(const char* string, bool case_sensitive, int
limit);
void SetHighlightTextMatches(bool highlight);
```

**Sample from
WebView**

OWB Control API



goForward

```
bool WebView::goForward()
{
    return m_page->goForward();
}
```

OWB Control API



goForward

```
bool WebView::goForward()  
{  
    return m_page->goForward();  
}
```

OWB Control API



Page::goForward

```
bool Page::goForward()
{
    HistoryItem* item = m_backForwardList->forwardItem();

    if (item) {
        goToItem(item, FrameLoadTypeForward);
        return true;
    }
    return false;
}
```

OWB Control API



Page::goForward

```
bool Page::goForward()
{
    HistoryItem* item = m_backForwardList->forwardItem();

    if (item) {
        goToItem(item, FrameLoadTypeForward);
        return true;
    }
    return false;
}
```

OWB Control API



Page::goToItem

```
void Page::goToItem(HistoryItem* item, FrameLoadType type)
{
    // Abort any current load if we're going to a history
    item
    m_mainFrame->loader()->stopAllLoaders();
    m_mainFrame->loader()->goToItem(item, type);
}
```



OWB Control API

WebKit

Optimizations

AddOn engine

Javascript API

Javascript API

AddOns
Implementation

AddOns
Implementation

WKAL

Private BAL

Graphics

Network

Types

OWBAL

...

Middleware, Libs, OS...



OWB Control API

WebKit

Optimizations

AddOn engine

Javascript API

Javascript API

AddOns
Implementation

AddOns
Implementation

WKAL

Private BAL

OWBAL

Graphics

Network

Types

...

Middleware, Libs, OS...

OWBAL



- Generated Abstraction Layer
- Turns Platform directory into a modular, independent and self-tested layer
- Fine-grained compilation options
- Self-tested API

OWBAL



- Refactoring of some WebKit files used “as is” in OWBAL
- Template skeleton of implementation for faster development

Generating OWBAL



- Intensive use of GCC-XML to cast and identify all functions, templates, signatures and moving them to the correct slot in OWBAL
- Separation of interface (skeleton) and implementation by modules then platform:
 - Database
 - GTK
 - WX...

Generating OWBAL



- Generation from GCC-XML of all internal data
- Done by tweaking CMAKE with gcc-xml as a compiler

```
cmake -DCMAKE_C_COMPILER="$OwnGCC" -DCMAKE_CXX_COMPILER="$OwnGcc"  
$OWBPATH  
make
```

- Results in .xml instead of .o with extensive description of original sources definition

Generating OWBAL



Widget.h preprocessing:

```
<Namespace id="_128" name="WebCore" context="_1" members="_526 _527 _528
_529 _530 _531 _532 _533 _540 _541 _545 _550 _551 _555 _556 "
mangled="_Z7WebCore" demangled="WebCore"/>
  <Class id="_541" name="Widget" context="_128"
mangled="N7WebCore6WidgetE" demangled="WebCore::Widget" location="f11:77"
file="f11" line="77" artificial="1" size="64" align="32" members="_717
_718 _719 _720 _721 _722 _723 _724 _725 _726 _727 _728 _729 _730 _731
_732 _733 _734 _735 _736 _737 _738 _739 _740 _741 _742 _743 _744 _745
_746 _747 _748 _749 _750 _751 _752 _753 _754 _755 _756 _757 _758 _759
_760 _761 _762 " bases="" />
    <Constructor id="_718" name="Widget" explicit="1" context="_541"
access="public" mangled="_ZN7WebCore6WidgetC1Ev *INTERNAL* "
demangled="WebCore::Widget::Widget()" location="f11:79" file="f11"
line="79" extern="1"/>
    <Destructor id="_719" name="Widget" virtual="1" context="_541"
access="public" mangled="_ZN7WebCore6WidgetD1Ev *INTERNAL* "
demangled="WebCore::Widget::~~Widget()" location="f11:80" file="f11"
line="80" extern="1">
```

Generating OWBAL



Widget.h preprocessing:

```
<Namespace id="_128" name="WebCore" context="_1" members="_526 _527 _528
_529 _530 _531 _532 _533 _540 _541 _545 _550 _551 _555 _556 "
mangled="_ZN7WebCore" demangled="WebCore"/>
<Class id="_541" name="Widget" context="_128"
mangled="N/WebCore6WidgetE" demangled="WebCore::Widget" location="f11:77"
file="f11" line="77" artificial="1" size="64" align="32" members="_717
_718 _719 _720 _721 _722 _723 _724 _725 _726 _727 _728 _729 _730 _731
_732 _733 _734 _735 _736 _737 _738 _739 _740 _741 _742 _743 _744 _745
_746 _747 _748 _749 _750 _751 _752 _753 _754 _755 _756 _757 _758 _759
_760 _761 _762 " bases="" />
  <Constructor id="_718" name="Widget" explicit="1" context="_541"
access="public" mangled="_ZN7WebCore6WidgetC1Ev *INTERNAL* "
demangled="WebCore::Widget::Widget()" location="f11:79" file="f11"
line="79" extern="1"/>
  <Destructor id="_719" name="Widget" virtual="1" context="_541"
access="public" mangled="_ZN7WebCore6WidgetD1Ev *INTERNAL* "
demangled="WebCore::Widget::~~Widget()" location="f11:80" file="f11"
line="80" extern="1">
```

Generating OWBAL



Widget.h preprocessing:

```
<Namespace id="_128" name="WebCore" context="_1" members="_526 _527 _528
_529 _530 _531 _532 _533 _540 _541 _545 _550 _551 _555 _556 "
mangled="_Z7WebCore" demangled="WebCore"/>
  <Class id="_541" name="Widget" context="_128"
mangled="N7WebCore6WidgetE" demangled="WebCore::Widget" location="f11:77"
file="f11" line="77" artificial="1" size="64" align="32" members="_717
_718 _719 _720 _721 _722 _723 _724 _725 _726 _727 _728 _729 _730 _731
_732 _733 _734 _735 _736 _737 _738 _739 _740 _741 _742 _743 _744 _745
_746 _747 _748 _749 _750 _751 _752 _753 _754 _755 _756 _757 _758 _759
_760 _761 _762 " bases="" />
    <Constructor id="_718" name="Widget" explicit="1" context="_541"
access="public" mangled="_ZN7WebCore6WidgetC1Ev *INTERNAL* "
demangled="WebCore::Widget::Widget()" location="f11:79" file="f11"
line="79" extern="1"/>
    <Destructor id="_719" name="Widget" virtual="1" context="_541"
access="public" mangled="_ZN7WebCore6WidgetD1Ev *INTERNAL* "
demangled="WebCore::Widget::~~Widget()" location="f11:80" file="f11"
line="80" extern="1">
```

Generating OWBAL



Widget.h preprocessing:

```
<Namespace id="_128" name="WebCore" context="_1" members="_526 _527 _528
_529 _530 _531 _532 _533 _540 _541 _545 _550 _551 _555 _556 "
mangled="_Z7WebCore" demangled="WebCore"/>
  <Class id="_541" name="Widget" context="_128"
mangled="N7WebCore6WidgetE" demangled="WebCore::Widget" location="f11:77"
file="f11" line="77" artificial="1" size="64" align="32" members="_717
_718 _719 _720 _721 _722 _723 _724 _725 _726 _727 _728 _729 _730 _731
_732 _733 _734 _735 _736 _737 _738 _739 _740 _741 _742 _743 _744 _745
_746 _747 _748 _749 _750 _751 _752 _753 _754 _755 _756 _757 _758 _759
_760 _761 _762 " bases="" />
    <Constructor id="_718" name="Widget" explicit="1" context="_541"
access="public" mangled="_ZN7WebCore6WidgetC1Ev *INTERNAL* "
demangled="WebCore::Widget::Widget()" location="f11:79" file="f11"
line="79" extern="1"/>
    <Destructor id="_719" name="Widget" virtual="1" context="_541"
access="public" mangled="_ZN7WebCore6WidgetD1Ev *INTERNAL* "
demangled="WebCore::Widget::~~Widget()" location="f11:80" file="f11"
line="80" extern="1">
```


Generating OWBAL



Widget.h result:

```
/**
 * \file Widget.h
 * \author WebKit project, modified by PLEYO team
 * \version svn_32291
 * \date 2008-04-24
 */
/**
 * \namespace WKAL
 * \brief WebKit part of the BAL
 */
namespace WKAL{
    /**
     * \class Widget
     * \brief
     *
     */
    class Widget{
    public:
        /**
         * \brief
         */
        Widget();
        /**
         * \brief
         */
        ~Widget();
        /**
         *
         * \param[in] long* someArg :
         * \return int :
         */
        int foobar(long* someArg);
    };
};
```

Generating OWBAL



Widget.h result:

```
/**
 * \file Widget.h
 * \author WebKit project, modified by PLEYO team
 * \version svn_32291
 * \date 2008-04-24
 */
/**
 * \namespace WKAL
 * \brief WebKit part of the BAL
 */
namespace WKAL{
    /**
     * \class Widget
     * \brief
     *
     */
    class Widget{
    public:
        /**
         * \brief
         */
        Widget();
        /**
         * \brief
         */
        ~Widget();
        /**
         *
         * \param[in] long* someArg :
         * \return int :
         */
        int foobar(long* someArg);
    };
};
```

Doxygen-ready

Generating OWBAL



Widget.h result:

```
/**
 * \file Widget.h
 * \author WebKit project, modified by PLEYO team
 * \version svn_32291
 * \date 2008-04-24
 */
/**
 * \namespace WKAL
 * \brief WebKit part of the BAL
 */
namespace WKAL{
    /**
     * \class Widget
     * \brief
     *
     */
    class Widget{
    public:
        /**
         * \brief
         */
        Widget();
        /**
         * \brief
         */
        ~Widget();
        /**
         *
         * \param[in] long* someArg :
         * \return int :
         */
        int foobar(long* someArg);
    };
};
```

Fill the blanks !

Generating OWBAL



Platform

x.h

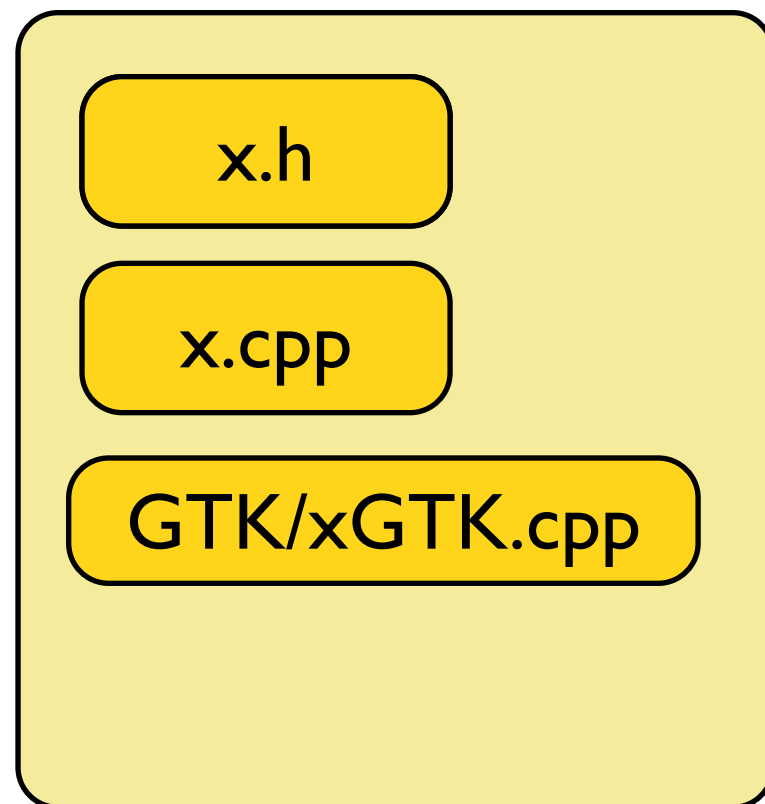
x.cpp

GTK/xGTK.cpp

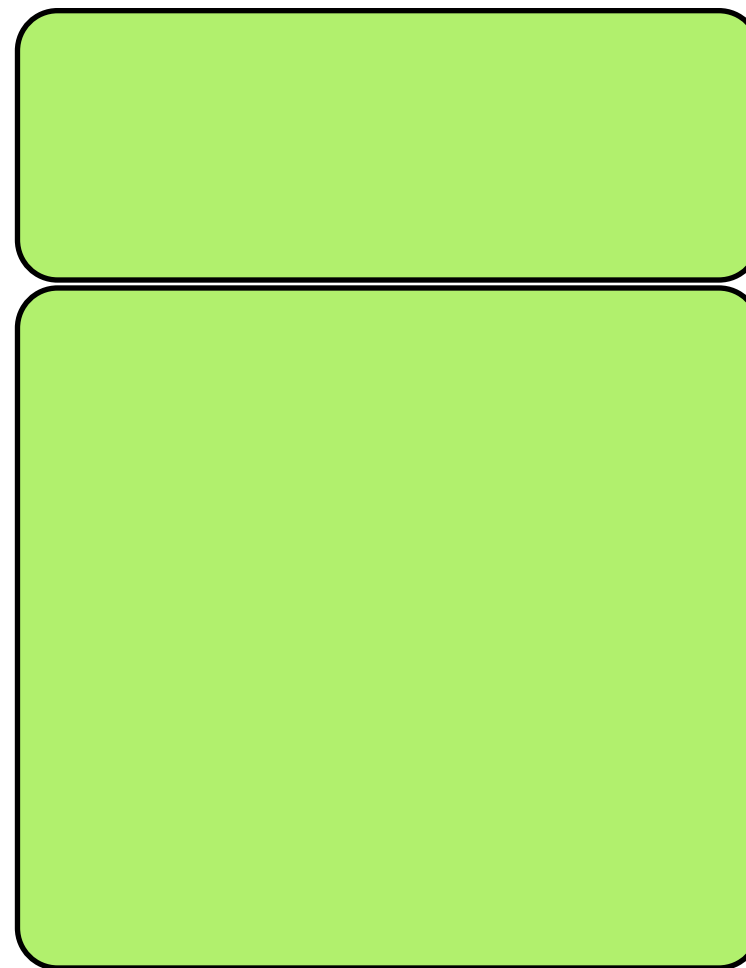
Generating OWBAL



Platform



OWBAL



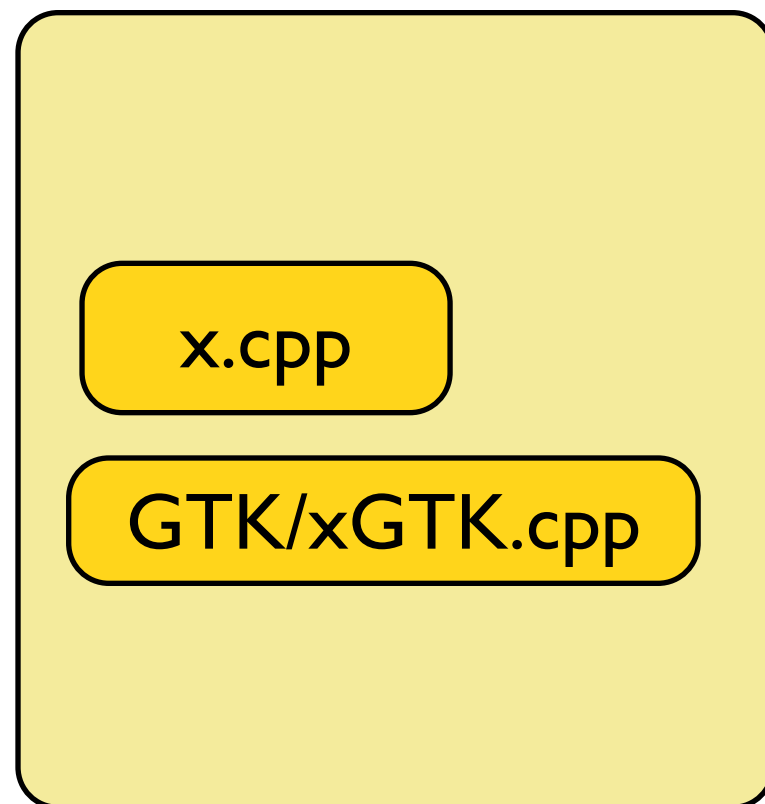
Skeleton

Concretization

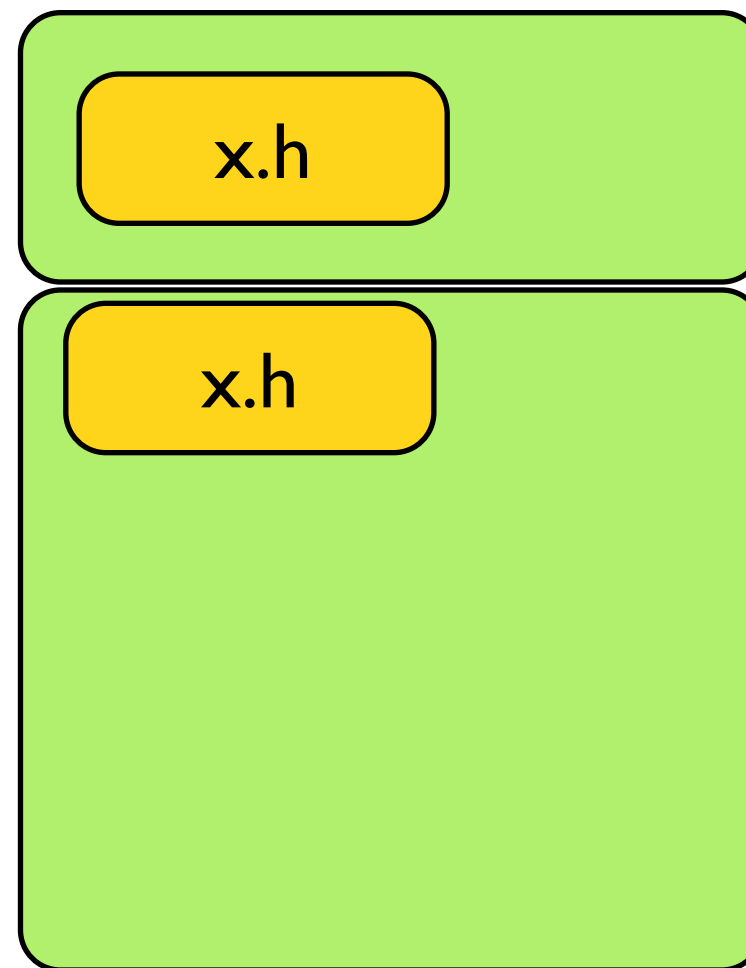
Generating OWBAL



Platform



OWBAL



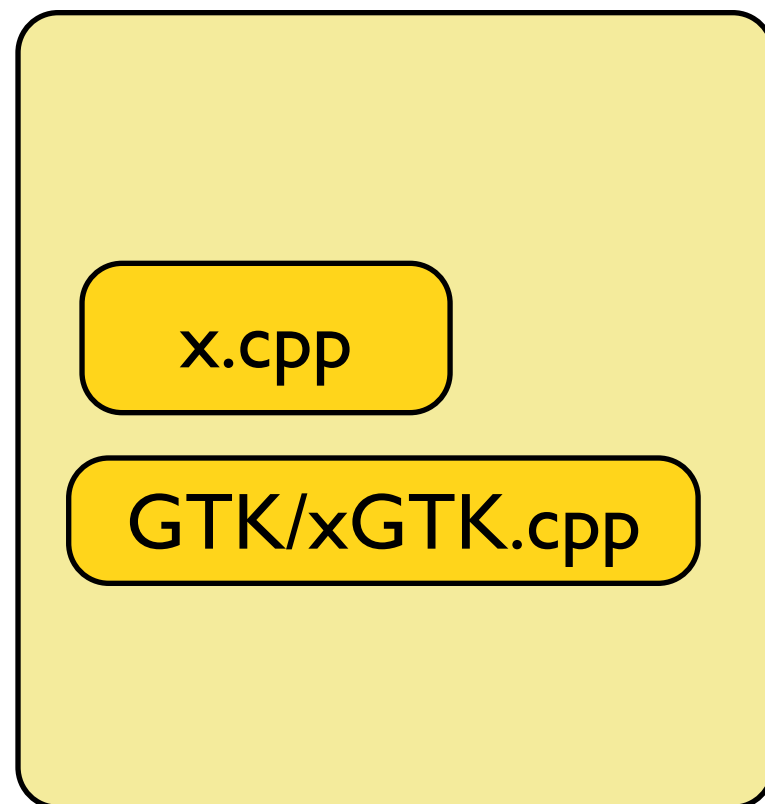
Skeleton

Concretization

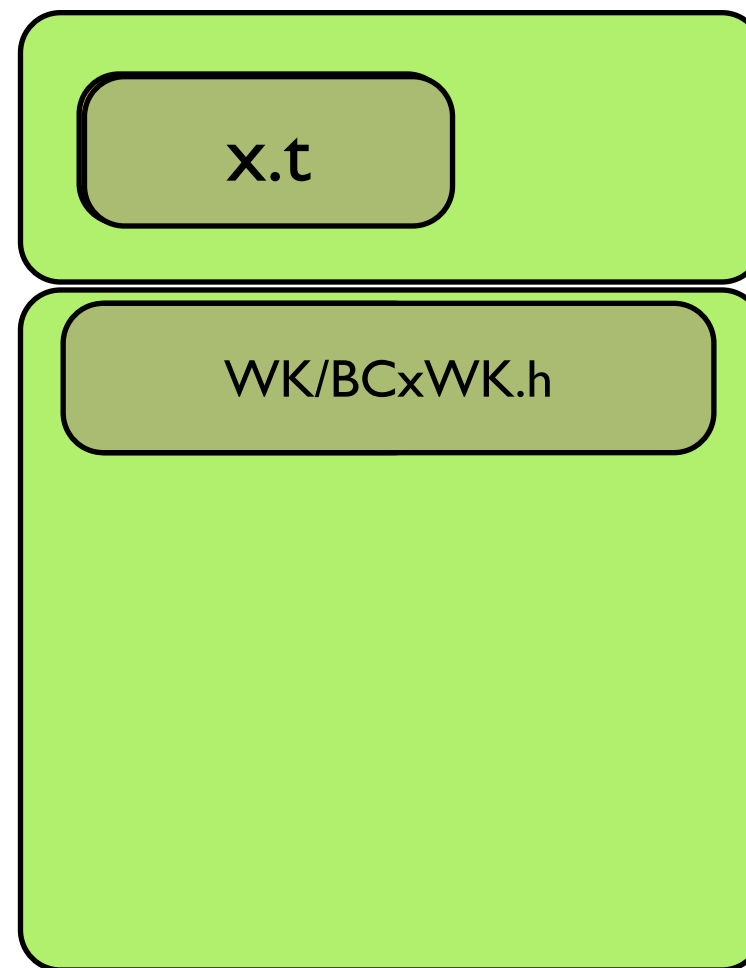
Generating OWBAL



Platform



OWBAL



Skeleton

Concretization

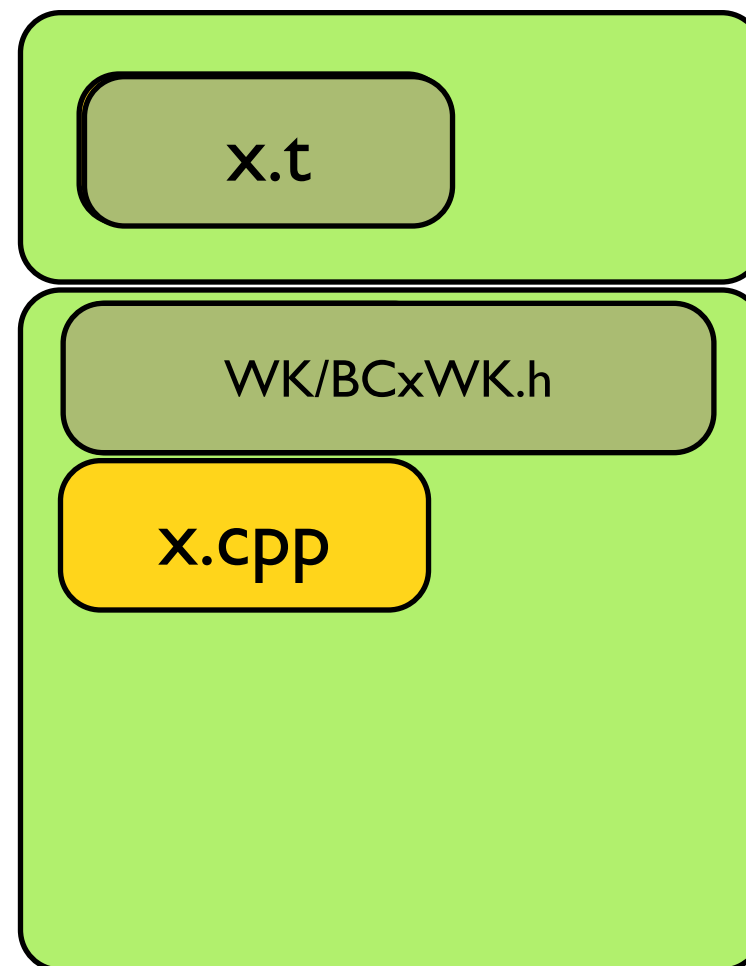
Generating OWBAL



Platform



OWBAL



Skeleton

Concretization

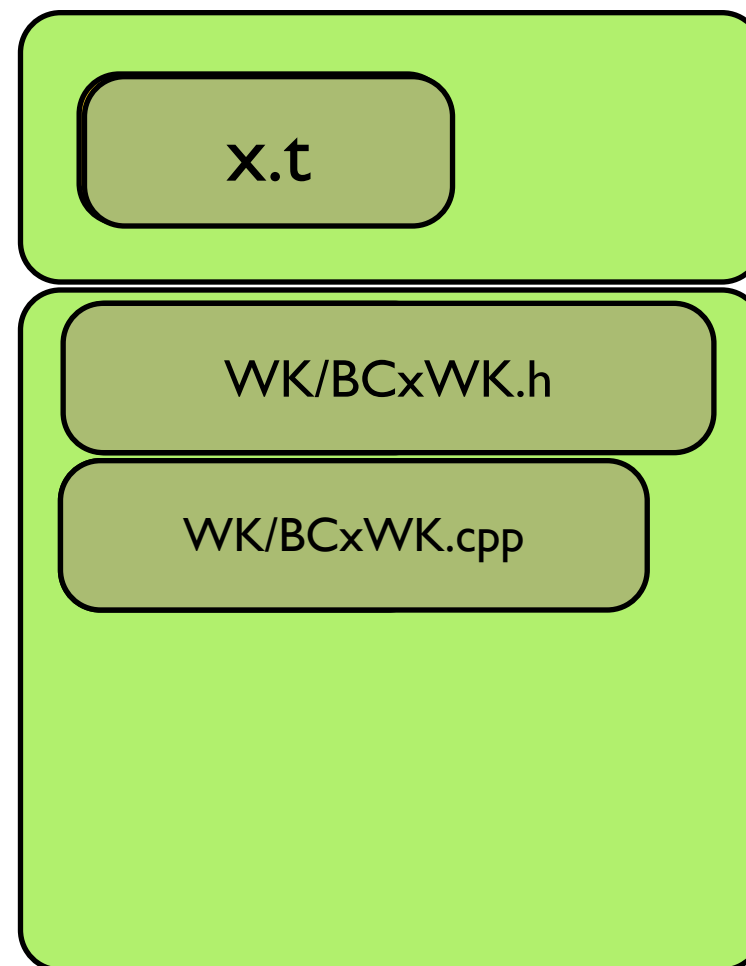
Generating OWBAL



Platform



OWBAL



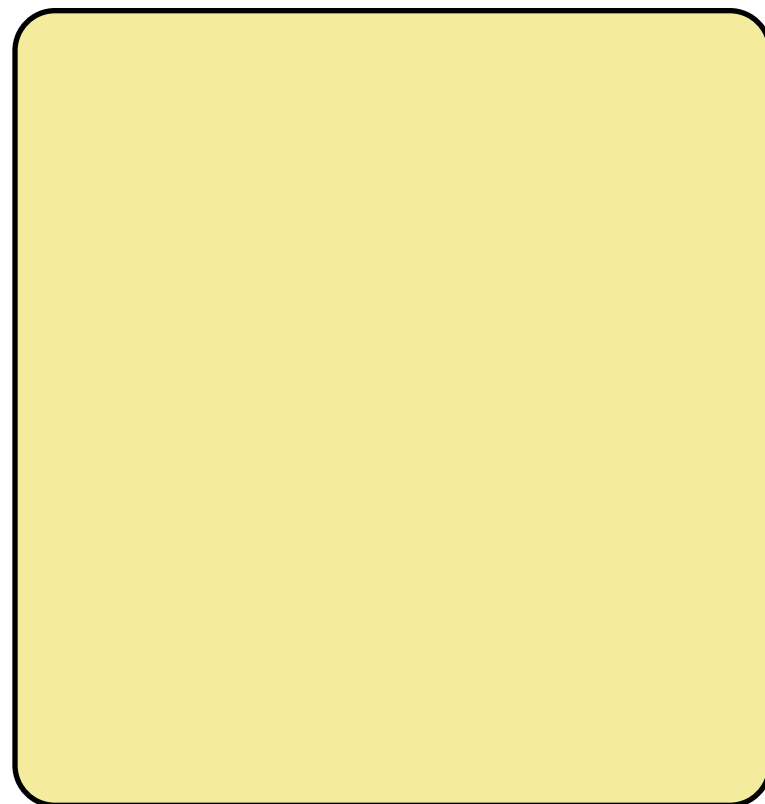
Skeleton

Concretization

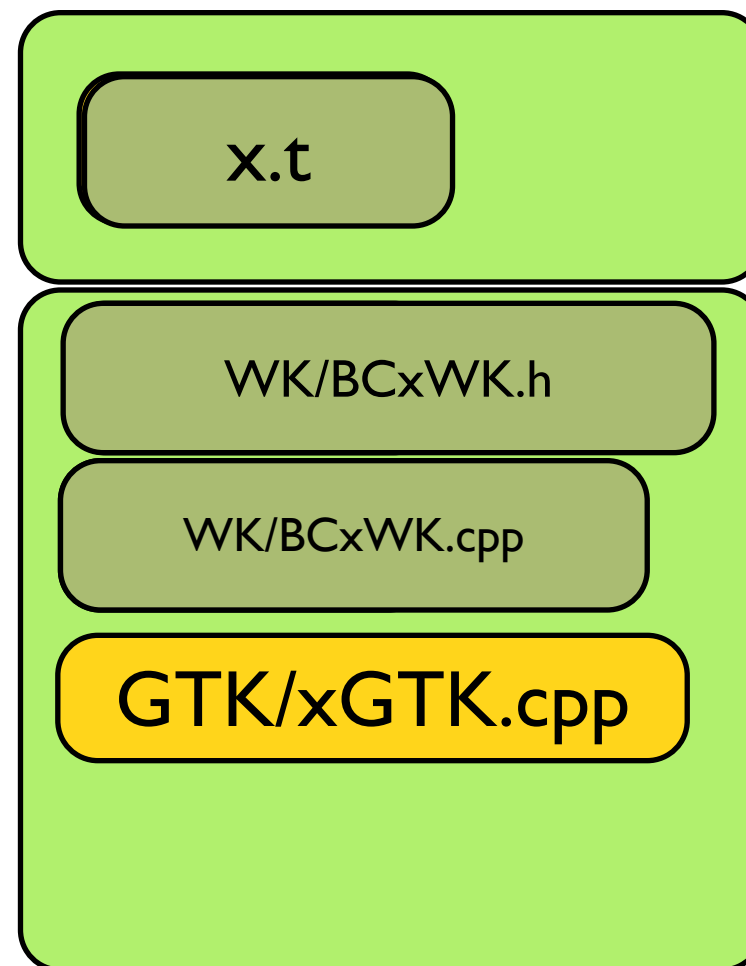
Generating OWBAL



Platform



OWBAL



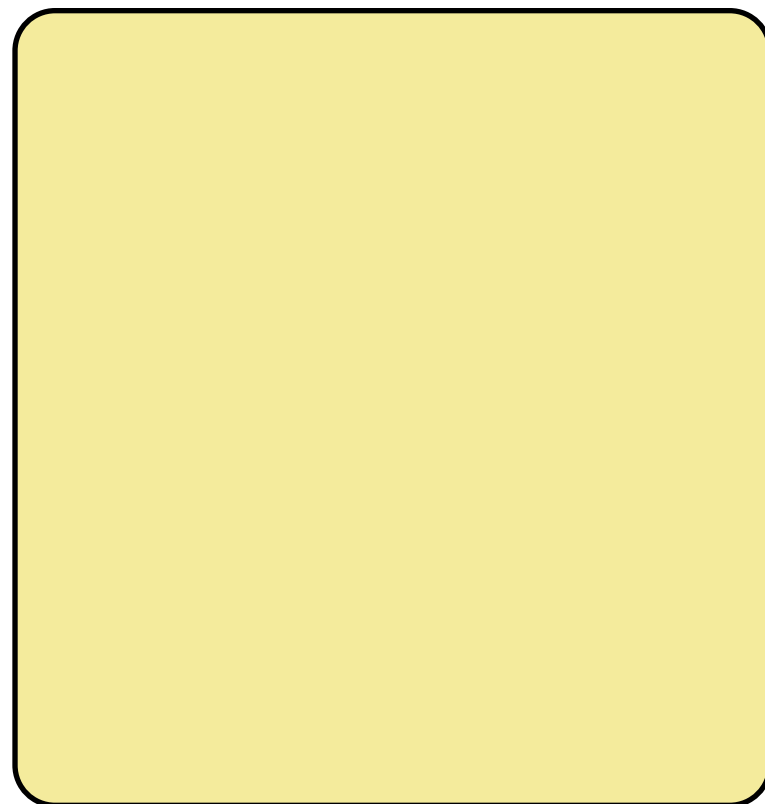
Skeleton

Concretization

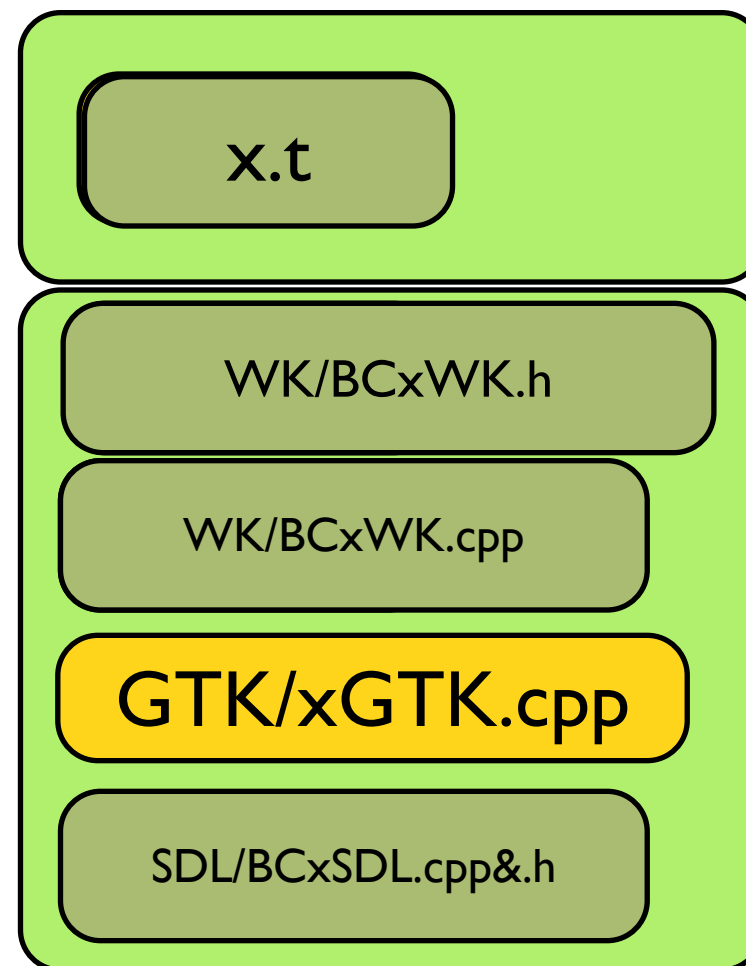
Generating OWBAL



Platform



OWBAL



Skeleton

Concretization

Generating OWBAL



Concretization of Widget for SDL:

- BCWidgetSDL.h
- BCWidgetSDL.cpp



```
/**
 * \file BCWidgetSDL.h
 * \version svn_32291
 * \date 2008-04-24
 */
#ifndef BCWIDGETSDL_H
#define BCWIDGETSDL_H

/**
 * \namespace WKAL
 * \brief WebKit par of the BAL
 */
namespace WKAL{
    /**
     * \class Widget
     * \brief This class handles things...
     */
    class Widget{
    public:
        /**
         * \brief Creates a default Widget
         */
        Widget();
        /**
         * \brief Destruct the Widget and remove links to its parents.
         */
        ~Widget();
        /**
         * \brief Makes a good foobar
         * Takes someArg and play with it to produce the right int we want to return ;)
         * \param[in] long* someArg : some random number
         * \return int : a number
         */
        int foobar(long* someArg);
    };//end of class Widget
};//end of namespace WKAL
//BCWIDGETSDL_H
#endif
```



```
/**
 * \file BCWidgetSDL.h
 * \version svn_32291
 * \date 2008-04-24
 */
#ifndef BCWIDGETSDL_H
#define BCWIDGETSDL_H
```

```
/**
 * \namespace WKAL
 * \brief WebKit par of the BAL
 */
```

```
namespace WKAL{
```

```
    /**
     * \class Widget
     * \brief This class handles things...
     */
```

```
    class Widget{
```

```
    public:
```

```
        /**
         * \brief Creates a default Widget
         */
```

```
        Widget();
```

```
        /**
         * \brief Destruct the Widget and remove links to its parents.
         */
```

```
        ~Widget();
```

```
        /**
```

```
         * \brief Makes a good foobar
```

```
         * Takes someArg and play with it to produce the right int we want to return ;)
```

```
         * \param[in] long* someArg : some random number
```

```
         * \return int : a number
```

```
         */
```

```
        int foobar(long* someArg);
```

```
    };//end of class Widget
```

```
};//end of namespace WKAL
```

```
//BCWIDGETSDL_H
```

```
#endif
```

BCWidgetSDL.h



```
/**
 * \file BCWidgetSDL.cpp
 * \version svn_32291
 * \date 2008-04-24
 */
#include "BCWidget.h"
namespace WKAL{
Widget::Widget()
{
}
Widget::~Widget()
{
}
int Widget::foobar(long* someArg)
{
}
;
```



```
/**
 * \file BCWidgetSDL.cpp
 * \version svn_32291
 * \date 2008-04-24
 */
#include "BCWidget.h"
namespace WKAL{
Widget::Widget()
{
}
Widget::~Widget()
{
}
int Widget::foobar(long* someArg)
{
}
;
```

BCWidgetSDL.cpp

Testing OWBAL



- Since OWBAL defines a single level API, testing is easier
- Black box testing of the API
- White box testing of the implementation

Testing OWBAL



```
namespace OWBAL {  
  
    class SQLValue {  
    public:  
        enum Type { NullValue, NumberValue, StringValue };  
  
        SQLValue() : m_type(NullValue) { }  
        SQLValue(double number) : m_type(NumberValue), m_number(number) { }  
        SQLValue(const String& s) : m_type(StringValue), m_string(s) { }  
        SQLValue(const SQLValue&);  
  
        Type type() const { return m_type; }  
  
        String string() const;  
        double number() const;  
  
    private:  
        Type m_type;  
        double m_number;  
        String m_string;  
    };  
};
```

BCSQLValueSQLite.h

Testing OWBAL



```
#ifndef SQLVALUETEST_H
#define SQLVALUETEST_H

#include <cppunit/extensions/HelperMacros.h>
#include "SQLValue.h"

class SQLValueTest : public CppUnit::TestFixture
{
    CPPUNIT_TEST_SUITE( SQLValueTest );
    CPPUNIT_TEST( testConstructor );
    CPPUNIT_TEST( testConstructorStress );
    CPPUNIT_TEST_SUITE_END();

public:
    void setUp();
    void tearDown();

    void testConstructor();
    void testConstructorStress();
};

#endif // MONEYTEST_H
```

SQL Value Unit Test

Testing OWBAL



```
void SQLValueTest::testConstructor()
{
    // Set up
    // Process
    SQLValue sqlValue;
    SQLValue sqlValue1(1.2);
    SQLValue sqlValue2("value");
    SQLValue sqlValue3(sqlValue1);

    // Check
    CPPUNIT_ASSERT_EQUAL( SQLValue::NullValue, sqlValue.type() );
    CPPUNIT_ASSERT_EQUAL( SQLValue::NumberValue, sqlValue1.type() );
    CPPUNIT_ASSERT_EQUAL( SQLValue::StringValue, sqlValue2.type() );
    CPPUNIT_ASSERT_EQUAL( SQLValue::NumberValue, sqlValue3.type() );
    CPPUNIT_ASSERT_EQUAL( 1.2, sqlValue1.number() );
    CPPUNIT_ASSERT( "value" == sqlValue2.string() );
    CPPUNIT_ASSERT_EQUAL( 1.2, sqlValue3.number() );
}
```

**SQL Value Unit Test:
testConstructor**

Testing OWBAL



```
void SQLValueTest::testConstructorStress()
{
    // Set up

    // Process & Check
    for(int i=0; i<1000; i++) {
        SQLValue *sqlValue = new SQLValue();
        SQLValue *sqlValue1 = new SQLValue(1.2);
        SQLValue *sqlValue2 = new SQLValue("value");
        SQLValue *sqlValue3 = new SQLValue(*sqlValue1);

        delete sqlValue;
        delete sqlValue1;
        delete sqlValue2;
        delete sqlValue3;
    }

    //Check
}
```

**SQL Value Unit Test:
testConstructor Stress**

OWBAL Compilation



- Fine-grained compilation options
 - Leverages CMake power
 - Cut any single WebKit feature (JS, SVG, ...)
 - Choose underlying library, switch from an implementation to another

OWBAL Compilation



```
Page 1 of 1
BISON_EXECUTABLE      /usr/bin/bison
BUILD_SHARED_LIBS     ON
CMAKE_BACKWARDS_COMPATIBILITY 2.4
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX  /usr/local
CMAKE_OSX_ARCHITECTURES i386
CMAKE_OSX_SYSROOT      /Developer/SDKs/MacOSX10.4u.sdk
COMPILE_TESTS         ON
DEEPSEE_ENABLE        OFF
DEEPSEE_USE_FAKED_DEEPSEE ON
DEEPSEE_USE_FAKED_DEEPSEE_PATH BAL/Includes/FakedDeepsee.h
EXECUTABLE_OUTPUT_PATH
FLEX_EXECUTABLE       /usr/bin/flex
GPERF_EXECUTABLE      /usr/bin/gperf
LIBRARY_OUTPUT_PATH
N800_BUILD            OFF
OWBAL_PLATFORM_GTK    GTK
OWBAL_PLATFORM_MACPORT ON
OWB_FAILSAFE_WITHOUT_DEEPSEE ON
WEBKIT_USE_CC_NO_STACK_PROTECT OFF
WEBKIT_USE_CC_PIC     OFF
WEBKIT_USE_CC_RTTI    OFF
WEBKIT_USE_JS         ON
WEBKIT_USE_NPAPI      OFF
WEBKIT_USE_UCLIBC     OFF
WEBKIT_USE_WEB_UI     OFF
WEBKIT_USE_XML_SUPPORT LIBXML
WEBKIT_USE_XSLT_SUPPORT LIBXSLT
```

Disable JavascriptCore

OWBAL Compilation



Page 1 of 1

BISON_EXECUTABLE	/usr/bin/bison
BUILD_SHARED_LIBS	ON
CMAKE_BACKWARDS_COMPATIBILITY	2.4
CMAKE_BUILD_TYPE	
CMAKE_INSTALL_PREFIX	/usr/local
CMAKE_OSX_ARCHITECTURES	i386
CMAKE_OSX_SYSROOT	/Developer/SDKs/MacOSX10.4u.sdk
COMPILE_TESTS	ON
DEEPSEE_ENABLE	OFF
DEEPSEE_USE_FAKED_DEEPSEE	ON
DEEPSEE_USE_FAKED_DEEPSEE_PATH	BAL/Includes/FakedDeepsee.h
EXECUTABLE_OUTPUT_PATH	
FLEX_EXECUTABLE	/usr/bin/flex
GPERF_EXECUTABLE	/usr/bin/gperf
LIBRARY_OUTPUT_PATH	
N800_BUILD	OFF
OWBAL_PLATFORM_GTK	GTK
OWBAL_PLATFORM_MACPORT	ON
OWB_FAILSAFE_WITHOUT_DEEPSEE	ON
WEBKIT_USE_CC_NO_STACK_PROTECT	OFF
WEBKIT_USE_CC_PIC	OFF
WEBKIT_USE_CC_RTTI	OFF
WEBKIT_USE_JS	ON
WEBKIT_USE_NPAPI	OFF
WEBKIT_USE_UCLIBC	OFF
WEBKIT_USE_WEB_UI	OFF
WEBKIT_USE_XML_SUPPORT	LIBXML
WEBKIT_USE_XSLT_SUPPORT	LIBXSLT

Compile using μ cLibC

OWBAL Compilation



```
Page 1 of 1
BISON_EXECUTABLE      /usr/bin/bison
BUILD_SHARED_LIBS     ON
CMAKE_BACKWARDS_COMPATIBILITY 2.4
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX  /usr/local
CMAKE_OSX_ARCHITECTURES i386
CMAKE_OSX_SYSROOT      /Developer/SDKs/MacOSX10.4u.sdk
COMPILE_TESTS         ON
DEEPSEE_ENABLE        OFF
DEEPSEE_USE_FAKED_DEEPSEE ON
DEEPSEE_USE_FAKED_DEEPSEE_PATH BAL/Includes/FakedDeepsee.h
EXECUTABLE_OUTPUT_PATH
FLEX_EXECUTABLE       /usr/bin/flex
GPERF_EXECUTABLE      /usr/bin/gperf
LIBRARY_OUTPUT_PATH
N800_BUILD            OFF
OWBAL_PLATFORM_GTK    GTK
OWBAL_PLATFORM_MACPORT ON
OWB_FAILSAFE_WITHOUT_DEEPSEE ON
WEBKIT_USE_CC_NO_STACK_PROTECT OFF
WEBKIT_USE_CC_PIC     OFF
WEBKIT_USE_CC_RTTI    OFF
WEBKIT_USE_JS         ON
WEBKIT_USE_NPAPI      OFF
WEBKIT_USE_UCLIBC     OFF
WEBKIT_USE_WEB_UI     OFF
WEBKIT_USE_XML_SUPPORT LIBXML
WEBKIT_USE_XSLT_SUPPORT LIBXSLT
```

Toggle XML Parser

OWBAL Compilation



Toggle Font Engine

WEBKIT_USE_CC_EXCEPTIONS	OFF
WEBKIT_USE_CC_NO_STACK_PROTECT	OFF
WEBKIT_USE_CC_PIC	ON
WEBKIT_USE_CC_RTTI	OFF
WEBKIT_USE_EVENTS	SDL
WEBKIT_USE_FONTS	FREETYPE
WEBKIT_USE_GRAPHICS	SDL
WEBKIT_USE_I18N	ICU
WEBKIT_USE_IMAGEDECODER	GENERIC
WEBKIT_USE_JS	ON
WEBKIT_USE_NPAPI	OFF
WEBKIT_USE_UCLIBC	OFF
WEBKIT_USE_WEB_UI	ON

OWBAL Compilation



Toggle Internationalization support

WEBKIT_USE_CC_EXCEPTIONS	OFF
WEBKIT_USE_CC_NO_STACK_PROTECT	OFF
WEBKIT_USE_CC_PIC	ON
WEBKIT_USE_CC_RTTI	OFF
WEBKIT_USE_FONTS	FREETYPE
WEBKIT_USE_GRAPHICS	SDL
WEBKIT_USE_I18N	ICU
WEBKIT_USE_IMAGEDECODER	GENERIC
WEBKIT_USE_JS	ON
WEBKIT_USE_NPAPI	OFF
WEBKIT_USE_UCLIBC	OFF
WEBKIT_USE_WEB_UI	ON

OWBAL Compilation



Toggle Image Decoder Libs

WEBKIT_USE_CC_EXCEPTIONS	OFF
WEBKIT_USE_CC_NO_STACK_PROTECT	OFF
WEBKIT_USE_CC_PIC	ON
WEBKIT_USE_EVENTS	OFF
WEBKIT_USE_FONTS	FREETYPE
WEBKIT_USE_GRAPHICS	SDL
WEBKIT_USE_I18N	ICU
WEBKIT_USE_IMAGEDECODER	GENERIC
WEBKIT_USE_JS	ON
WEBKIT_USE_NPAPI	OFF
WEBKIT_USE_UCLIBC	OFF
WEBKIT_USE_WEB_UI	ON

Metrics on ST 7109 ref design

Options	-DMOBILE=1 -DLOW_BANDWIDTH_DISPLAY=0	-DMOBILE=0 -DLOW_BANDWIDTH_DISPLAY=0	-DMOBILE=0 -DLOW_BANDWIDTH_DISPLAY=1	-DMOBILE=1 -DLOW_BANDWIDTH_DISPLAY=1
WebSite				
google.fr: best worst average	1.220 1.871 1.6	1.294 1.915 1.5	1.189 1.681 1.4	1.197 1.563 1.3 (static: 1.0)
maps.google.fr: best worst average	13.34 13.80 13.5	13.372 14.572 13.6	13.335 13.795 13.5	13.526 14.354 13.6
lemonde.fr: best worst average	18.96 23.70 21.8	17.776 79.140 46.5	18.940 76.909 36.5	24.113 27.617 26.2 (static: 17.2)
deviantart.com: best worst average	10.45 10.84 10.7	11.167 11.601 11.5	10.849 11.934 11.3	10.861 12.879 11.2 (static: 8.7)
fr.yahoo.com: best worst average	9.10 9.99 9.4	8.415 8.940 8.7	8.400 8.605 8.6	9.602 10.963 9.8 (static: 6.5)
gentoo.org: best worst average	1.44 2.87 1.9	1.863 4.908 2.5	1.299 4.307 2.4	1.485 2.561 1.9 (static: 1.0)

Note that static links increase performances by 20%

Leveraging WebKit



- Clean and simple extension mechanism
- Leverage Device capabilities

Leveraging WebKit



- Adopted by CEM / ODM
- AmigaOS Official Web Browser
- Ported on QNX, Windows Mobile, OS 21, Nucleus, Many Embedded Linux flavors...

Leveraging WebKit



- Presentation extension:
 - Optimizes layout rendering with ad-hoc rules
 - Leverages specific capabilities with JS or HTML extensions



PLEYO

WEB-ENABLER
FOR CE DEVICES



 **origyn**
Suite



PLEYO
WEB-ENABLER
FOR CE DEVICES

Origyn Suite



- If you have a QVGA screen
 - It's not worth downloading a 2048*1025 px picture
 - Ask your RAM to store it
 - Ask your CPU to resize it
 - Display a 200*100 picture...



PLEYO

WEB-ENABLER
FOR CE DEVICES

Origyn Suite





PLEYO

WEB-ENABLER
FOR CE DEVICES

Origyn Suite



By Country	
France	
Home	
Radio Nova	>
Europe 1	>
France Inter	>
France Culture	>
SkyRock	>
France Info	>
Le Mouv	>

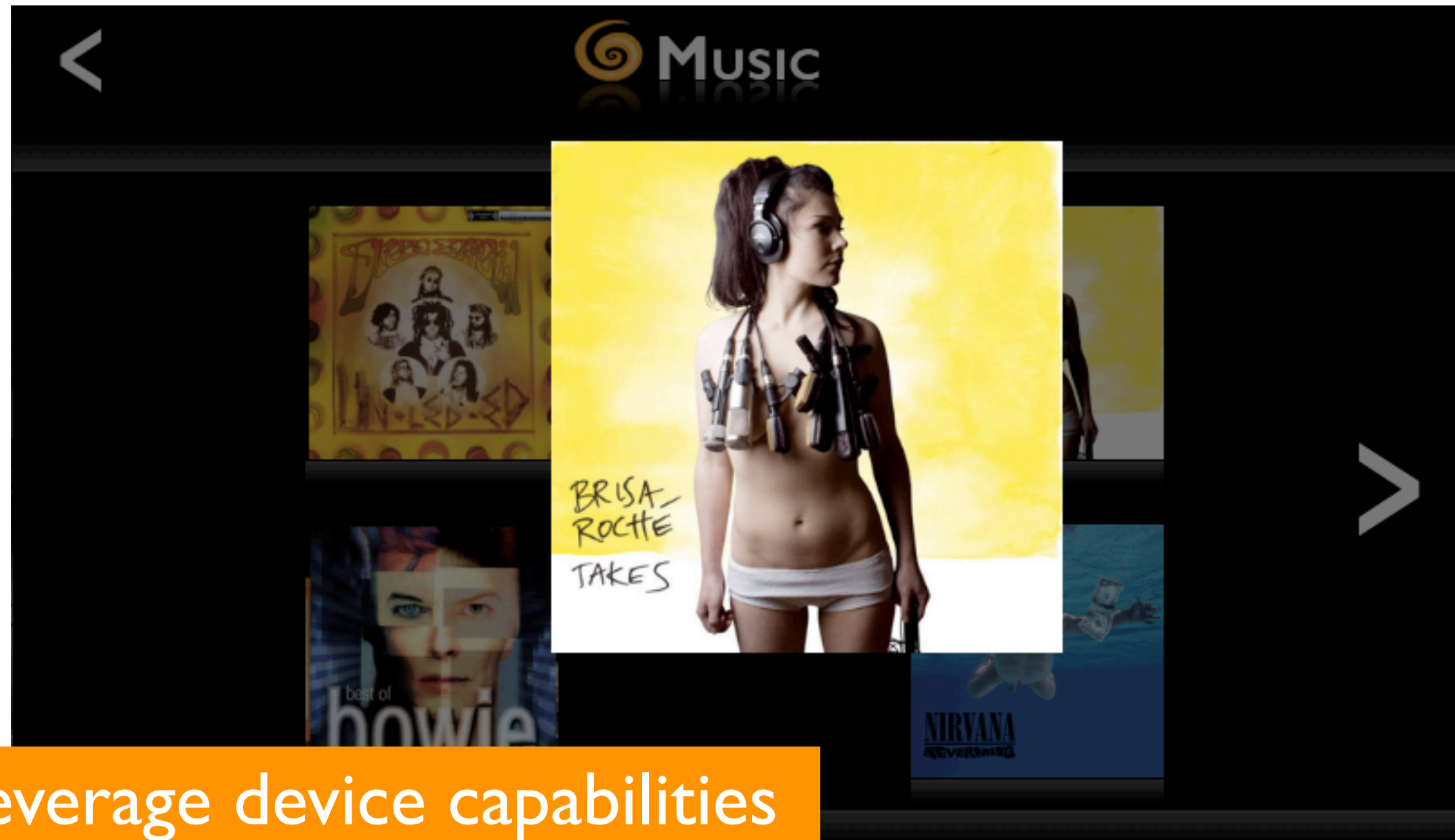
Use any look and feel



PLEYO

WEB-ENABLER
FOR CE DEVICES

Origyn Suite



Leverage device capabilities



PLEYO

WEB-ENABLER
FOR CE DEVICES

Origyn Suite



Leverage device capabilities



Some Screenshots...

Origyn Web Browser

Opensource
Web-Based Projects
for **CE Devices**



Illustration showing a stack of boxes. The top box is labeled "POST project" and the bottom box is labeled "VIEW projects". Below the boxes are two smaller boxes labeled "Press" and "News".



Illustration of a mobile phone (PDA) inside a box filled with gold coins, with the text "Sand-Labs.org" overlaid.

Sand-Labs.org

This website is intended to host projects aimed at bringing the **web technology to CE devices**.

The first hosted project is a refactor of WebKit for CE devices : OWB.



[About](#) [Contact](#) © 2007-Sand-labs.org

Linux SDL, FreeType fonts

Opensource
Web-Based Projects
for **CE Devices**



Press



News



This website is intended to host
projects aimed at bringing the
web technology to CE devices.

The first hosted project is a
refactor
of WebKit for CE devices : OWB.



[About](#) [Contact](#)

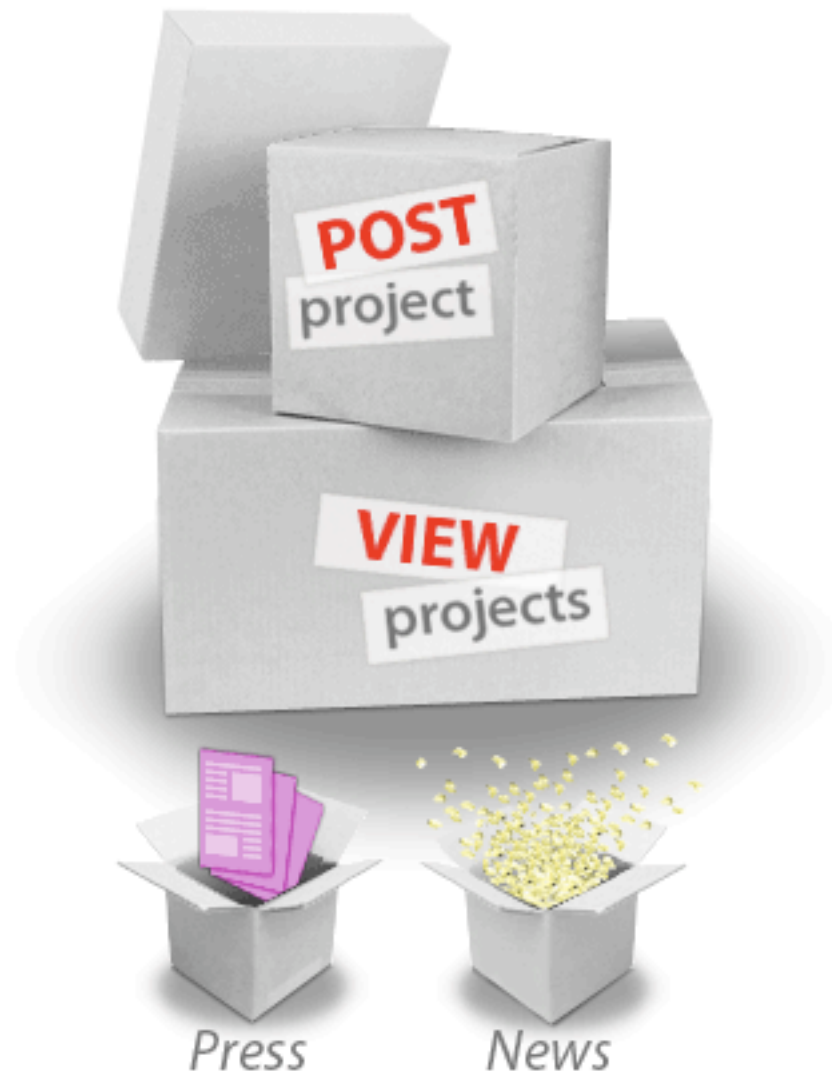
© 2007-Sand-labs.org

Linux SDL, Embedded Fonts





Opensource
Web-Based Projects
for **CE Devices**



This website is intended to host projects aimed at bringing the **web technology** to CE devices.

The first hosted project is a refactor of WebKit for CE devices : OWB.



[About](#) [Contact](#)

© 2007-Sand-labs.org

MacOS with MacPort libs
and GTK graphics

Web-Based Projects for CE Devices

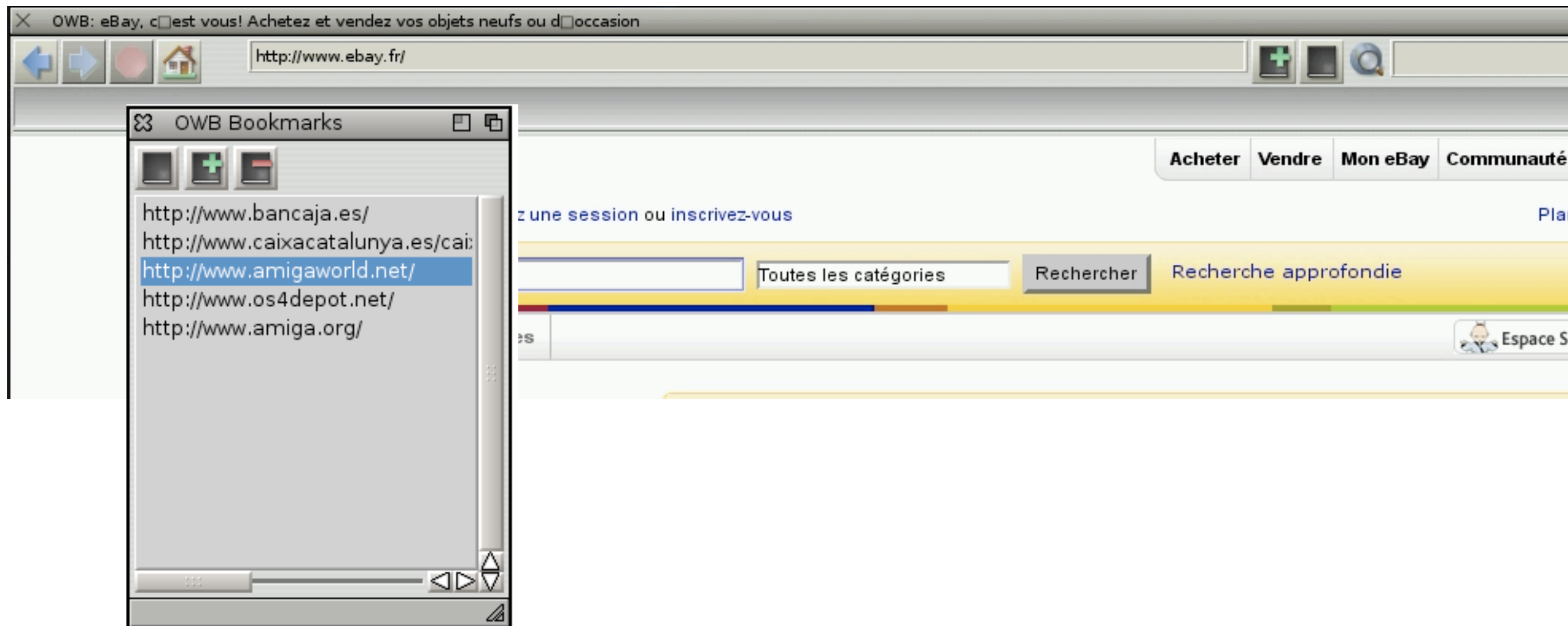
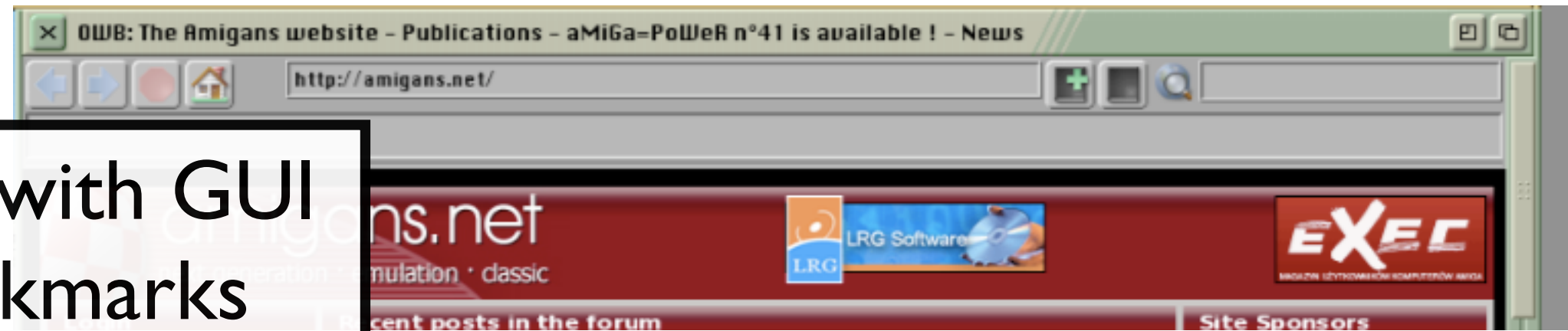


The first hosted project is a refactor of WebKit for CE devices : OWB.

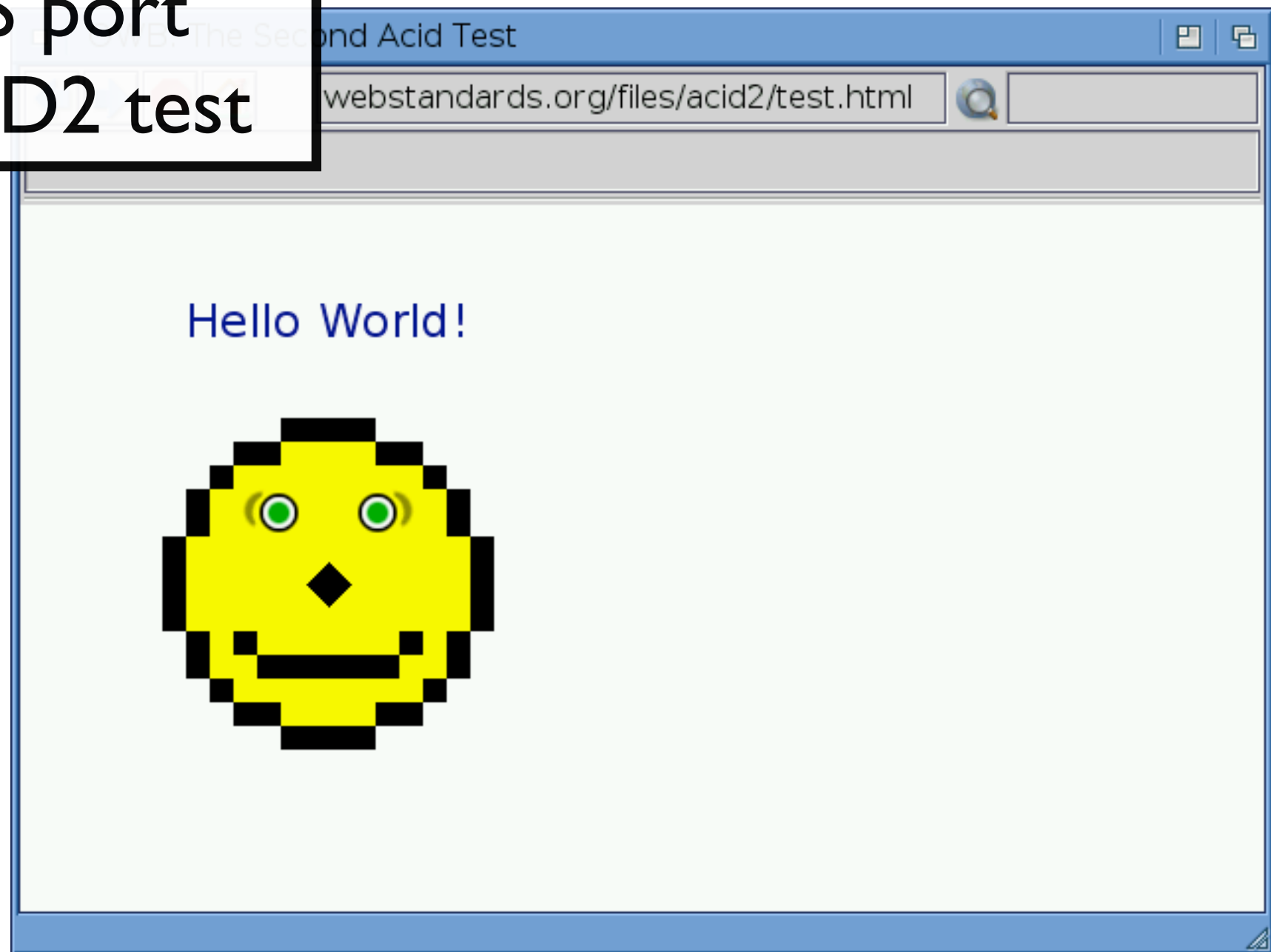


© 2007-Sand-lab s.org

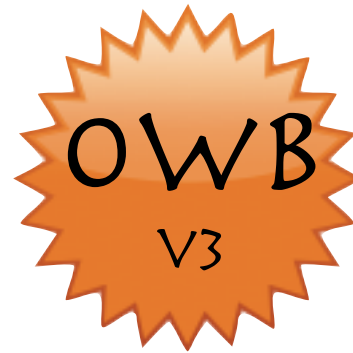
AmigaOS with GUI and bookmarks



AmigaOS port
passes ACID2 test

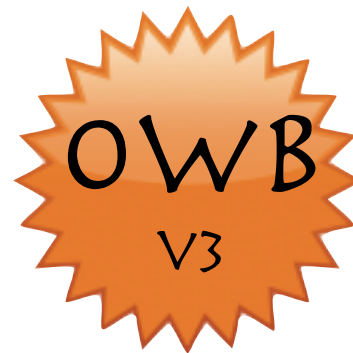








Available



Today

<http://www.sand-labs.org/svn/>



Get involved!

<http://www.sand-labs.org/owb>





Thank you



Questions?