

MayLoon User Manual

Copyright © 2013 Intel Corporation Document Number: xxxxxx-xxxUS World Wide Web: http://www.intel.com/software

Document Number: XXXXX-XXXXX



Disclaimer and Legal Information

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or by visiting Intel's Web Site.

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. See http://www.intel.com/products/processor_number for details.

BunnyPeople, Celeron, Celeron Inside, Centrino, Centrino Atom, Centrino Atom Inside, Centrino Inside, Centrino logo, Core Inside, FlashFile, i960, InstantIP, Intel, Intel logo, Intel386, Intel486, IntelDX2, IntelDX4, IntelSX2, Intel Atom, Intel Atom Inside, Intel Core, Intel Inside, Intel Inside logo, Intel. Leap ahead., Intel. Leap ahead. logo, Intel NetBurst, Intel NetMerge, Intel NetStructure, Intel SingleDriver, Intel SpeedStep, Intel StrataFlash, Intel Viiv, Intel vPro, Intel XScale, Itanium, Itanium Inside, MCS, MMX, Oplus, OverDrive, PDCharm, Pentium, Pentium Inside, skoool, Sound Mark, The Journey Inside, Viiv Inside, vPro Inside, VTune, Xeon, and Xeon Inside are trademarks of Intel Corporation in the U.S. and other countries.

* Other names and brands may be claimed as the property of others.

Copyright (C) 2010, Intel Corporation. All rights reserved.

Revision History

Document Number	Revision Number	Description	Revision Date
xxxxx	001	Initial release.	March 2013
ххххх	002	Updated for MayLoon alpha	May 2013
	003	Updated for MayLoon 0.3.1	June 2013
	004	Updated for MayLoon 0.4	July 2013
	005	Updated for MayLoon 0.5	August 2013

MayLoon User Manual



Table of Contents

1	Environment Requirements 4
2	Installing MayLoon Porting Tool 5
3	Uninstalling and Upgrading Mayloon Porting Tool83.1Uninstall3.2Upgrade9
4	Configuring MayLoon Porting Tool
5	Using MayLoon Porting Tool115.1Converting to MayLoon115.2Implementing Stubs165.3Building and Running185.4Reviewing and Editing JavaScript205.5Debuging Application in Browser245.6Packaging for Tizen27
6	Supported APIs
7	Unsupported Features
8	Known Incompatibilities and Limitations378.1Eclipse plugin378.2Incompatibility between Java and JavaScript37
9	Upcoming Features



1 Environment Requirements

- Linux or Windows PC
- Java JDK 1.6
- Eclipse Classical 3.7.2 (Indigo SR2) or higher with android ADT 21 or greater and JDT plugin installed
- MayLoon porting tool installation package (MPT.zip)
- MayLoon SDK archive file (MayLoon.zip)



2

Installing MayLoon Porting Tool

1. Open Eclipse IDE, Help-> Install New Software..., then click "Add..." button

Install	
Available Software Select a site or enter the location of a site.	
Work with: type or select a site	Find more software by working with the <u>Available Software Sites</u> preferences.
type filter text	
Name	Version
Select All Deselect All	
Details	14
Show only the latest versions of available software	Hide items that are already installed
Group items by category	What is <u>already installed</u> ?
Show only software applicable to target environment	t
Contact all update sites during install to find require	d software
?	< Back Next > Finish Cancel

Pic1 Install MPT

2. Click Archive..., then select MPT_xxx.zip and OK



Install	
Available Software Select a site or enter the location of a site.	
Work with: type or select a site Find r	Add more software by working with the <u>"Available Software Sites"</u> preferences.
type filter text	
Name	Version
Select All	Local MPT/MPT_0.3.zip!/ Archive OK Cancel
$\overline{ {oldsymbol V} }$ Show only the latest versions of available software	Hide items that are already installed
Group items by category	What is <u>already installed</u> ?
Show only software applicable to target environment	
Contact all update sites during install to find required soft	<pre>ware </pre> < Back Next > Finish Cancel

Pic2 Add installation package

3. Install MPT feature as normal eclipse plugin.

MayLoon User Manual



🔵 Install			- x
Available S Check the i	oftware tems that you wish to install.		
Work with:	MPT - jar:file:/C:/tools/mayloon/MPT/MPT_0. Find	3.zip!/	Add Sites" preferences.
type filter te	đ		
Name		Version	
D 🔽 000 N	ИРТ		
Select Al	Deselect All 1 item selected		
			-
Show only	the latest versions of available software	Hide items that are already installed	
👿 Group iter	ns by category	What is <u>already installed</u> ?	
Show only	/ software applicable to target environment		
Contact a	ll update sites during install to find required soft	ware	
?		< Back Next > Finish	Cancel

Pic3 Installation process

Note: If you get a security warning saying that the authenticity or validity of the software can't be established, click OK and continue.

Installing Software	Annan Coloma	
Security Warning		
Warning: You are installing software or validity of this software cannot be installation?	that contains unsigned content established. Do you want to co	t. The authenticity ntinue with the
	OK Cancel	Details >>
		J

Pic4 Security warning



3 Uninstalling and Upgrading Mayloon Porting Tool

3.1 Uninstall

1. Click Help->About Eclipse, then click Installation Details button in popup window:



Pic5 Uninstall 1

2. Select MayLoon porting tool plugin in Installed Software Tab and then click Uninstall... button:



lame	Version	Id	Provider
Android DDMS	20.0.0.v2012062420	com.android.ide.eclipse.ddms.feature.gro	The Android Open Source Proje
🖗 Android Development Tools	20.0.0.v2012062420	com.android.ide.eclipse.adt.feature.group	The Android Open Source Proje
🖗 Android Hierarchy Viewer	20.0.0.v2012062420	com.android.ide.eclipse.hierarchyviewer.f	The Android Open Source Proje
🖗 Android Traceview	20.0.0.v2012062420	com.android.ide.eclipse.traceview.feature	The Android Open Source Proje
Eclipse for RCP and RAP Developers	1.4.2.20120213-0814	epp.package.rcp	
🖗 MPT Feature	0.3.0	com.intel.ide.eclipse.mpt.feature.feature.g	INTEL
WTBot for Eclipse Forms Testing (incubation)	2.1.0.201302221524	org.eclipse.swtbot.forms.feature.group	Eclipse.org
WTBot for Eclipse Testing (incubation)	2.1.0.201302221524	org.eclipse.swtbot.eclipse.feature.group	Eclipse.org
WTBot for GEF Testing (incubation)	2.1.0.201302221524	org.eclipse.swtbot.eclipse.gef.feature.group	Eclipse.org
SWTBot for SWT Testing (incubation)	2.1.0.201302221524	org.eclipse.swtbot.feature.group	Eclipse.org
SWTBot IDE Features (incubation)	2.1.0.201302221524	org.eclipse.swtbot.ide.feature.group	Eclipse.org
SWTBot JUnit Headless launchers for Eclipse (incubation)	2.1.0.201302221524	org.eclipse.swtbot.eclipse.test.junit.featur	Eclipse.org
SWTBot Recorder and Test Generator (incubation)	2.1.0.201302221524	org.eclipse.swtbot.generator.feature.featu	Eclipse.org
Index Parties Teel			
ayloon Porting Tool			

Pic6 Uninstall 2

3. Then click Finish button to finish uninstalling and click Restart Now button to make uninstallation work immediately

3.2 Upgrade

If you want to Upgrade MayLoon porting tool, please uninstall installed MayLoon porting tool plugin as 3.1 describes, and then follow section 2 to install your new MayLoon porting tool.



4 Configuring MayLoon Porting Tool

Click Window->Preferences, select Mayloon on the left tab, there will be 1 item to configure:

SDK Location: Set proper location of MayLoon SDK (folder where MayLoon.zip is extracted) as the following pic shows

Click Apply and OK button to apply your settings.

Preferences	
type filter text	Mayloon $\Leftrightarrow \bullet \bullet \bullet \bullet \bullet \bullet$
General Android Ant Help Install/Update Java Mayloon Mylyn Plug-in Development Run/Debug SWTBot Preferences Team Usage Data Collector Validation XML	Mayloon Preferences SDK Location: C:\tools\mayloon\MPT\mayloonSDK Browse SDK version is 1.0. Support API Level 10
	Restore Defaults Apply
?	OK Cancel

Pic7 Mayloon preferences



5 Using MayLoon Porting Tool

This section goes through the steps to convert and build your Android application to HTML5.

5.1 Converting to MayLoon

- 1. Import an existing Android project into Eclipse
- 2. Build the Android project; make sure an android application package is generated successfully, usually under the bin/ directory of the Android project. Note if the application uses proprietary extensions to Android framework and cannot build in Eclipse environment, please generate an APK package corresponding to this application offline, and put it in the bin/ directory. The conversion cannot proceed without the APK file.
- 3. In Eclipse IDE, Select your project to convert, and then right click. Click "Convert to Mayloon Project".



Image: Second secon		Co Into	
 Com.intel.linpack.LinpackActivit Android 2.3.3 Android Dependencies 	6	Open in New Window Open Type Hierarchy Show In	F4 Alt+Shift+W ►
 Ben [Generated Java Files] Settings assets bin 		Copy Copy Qualified Name Paste Delete	Ctrl+C Ctrl+V Delete
 res .classpath .project AndroidManifest.xml 	<u>.</u>	Remove from Context Build Path Source	Ctrl+Alt+Shift+Down ► Alt+Shift+S ►
proguard.cfg project.properties	è	Refactor Import	Alt+Shift+T ►
	69 1	Build Project Refresh Close Project Assign Working Sets	F5
		Run As Debug As Validate	> >
		Convert to Mayloon Project	
		Team Compare With	+
		Android Tools	•

Pic8 Convert to Mayloon project

4. Convert Wizard will pop up, you can configure partial conversion mode for application.

Partial Conversion Mode: If partial conversion is enabled, MayLoon will modify application source code in the following way during conversion:

- It converts JNI method declarations into stub methods that you can later implement in either Java or JavaScript.
- Under this mode, MayLoon will also generate stub classes for the Android framework classes and 3rd party extensions that MayLoon does not support.

Select the checkbox to enable this mode. And you may also want to view these generated stub classes in eclipse after conversion, just select the checkbox titled "Review the created Java files after conversion".

Click Next> button to continue.



•	Convert to MayLoon Application	- 🗆 🗙
Convert Convert to MayLo	on Application	
Press "Next" to che	eck necessary files and information. onversion ted Java files after conversion	
?	< Back Next > Finish	Cancel

Pic9 Convert wizard

5. Check info will list in this page.

Error information: Errors that will terminate convert process

Warning information: Information that needs developer to pay attention to



•	
Check Before Convert	
8 Error and warning information shows below	
Error information:	
Can't find necessary Android file: C:\Share\source\mayloon-qa\AndroidAppSource\	\com.intel.jsdroid.calcula
4	
Warning information:	
Project depends on other library:C:\tools\mayloon\android app\com.intel.jsdroid.ca	alculator2\libs\arity-2.1.
	Þ
Kext > Finish	Cancel

Pic10 Check before conversion

Click Next> or Finish button if there is no error in project, otherwise please fix the error before converting project.

6. If partial conversion mode is enabled, stub class and method info will be listed. You can also click View Report button to view stub info in browser.

MayLoon User Manual



- the in the second sec	
Conversion Report	
Stub information shows below. All stub classes and methods shall be implemented man	ually.
Stub classes:	
Stub methods:	
View Report	
A Back Next > Finish	Cancel

Pic11 Conversion report

7. **Note** that after conversion, the project is still a Java project. When you run it as MayLoon project, it will be translated into an HTML5 application. After converting Android application to MayLoon project, if Partial Conversion mode is disabled, and converted application contains APIs/classes not supported by Mayloon, problem markers will be displayed in editor as the following pic shows:





Pic12 Converted MayLoon project without partial conversion

If Partial Conversion mode is enabled, stub classes/functions will be generated for classes not supported by Mayloon and JNI method declarations. Refer to section 5.2 for details. You can review the generated code before proceeding.

5.2 Implementing Stubs

If Partial Conversion mode is enabled, stub classes/functions will be generated for classes not supported by Mayloon and JNI method declarations. Here is a comparison between original android project and converted MayLoon project with partial conversion:



Pic13 Original Android project







A list of the generated stub methods and stub classes are printed in the Eclipse console. You can review the generated stubs and implement them:

• To implement the method add(int a, int b), please remove the comment including the @j2sNative just above the method, and implement the method using either Java code or JavaScript. Intuitively, the above method would be implemented as follows:



• The generated stub class EmbossMaskFilter contains 3 constructors that once called will error out. To implement the stub class, please implement all the necessary constructors and methods as in normal Android / Java application. If the stub class originates from Android framework, please refer to the official Android developer documentation - <u>http://developer.android.com/index.html</u> for detailed description of the class.

In addition to converting Java code to JavaScript, MayLoon also offers a hybrid programming model, meaning that you can embed JavaScript code into Java program, in the same way as you can embed assembly code into C/C++ program. MayLoon uses the Java2Script library to achieve hybrid programming. The @j2sNative block in Pic14 is an example illustrating how this works. Please refer to http://j2s.sourceforge.net/ for more information on hybrid programming using j2s.

Application developers can use hybrid programming for implementing stub classes and methods.



After the stubs are implemented as necessary, the project is ready to be translated into HTML5.

5.3 Building and Running

1. Finally, you can click "Run As" from the popup menu, and then click "Mayloon Application". If there are one or more main() entries in your application, **Select Java Application** dialog will pop up, select item **Start-android.core** and click OK to continue. In this step, all the Java source code files will be translated into JavaScript code, and an HTML file will be generated as the entry of converted web application.

▲ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓		New Go Into Open in New Window	Þ	
🕞 🗾 May				
⊳ 🕞 .sett		Open Type Hierarchy	F4	
🕞 asse		Show In	Alt+Shift+W ►	
⊳ 🗁 bin	B	Сору	Ctrl+C	
> 🗁 exte		Copy Qualified Name		
> 🗁 njs		Paste	Ctrl+V	
b 🗁 res		Delete	Delete	
⊳ 🗁 res_	^	Delete	Delete	-
X .clas	<u>_</u>	Remove from Context	Ctrl+Alt+Shift+Down	
		Build Path	+	
		Source	Alt+Shift+S ►	
🗷 arch		Refactor	Alt+Shift+T ▶	
📄 may				
📄 prog	è	Import		
📄 proj	4	Export		
		Build Project		Console & LogCat Ar Se
		Refresh	FS	5:693] [INFO] [Convert] Project 'com example
	\$	Class Desiret	15	2:318] [INFO] [Partial Conversion] Miss clas
		Close Project		2:359] [INFO] [Partial Conversion] Annotatio
		Assign Working Sets		[2:453] [INFO] [Convert] Project 'com.example 37:224] [INFO] [Convert] Project 'linn
		Run As	۱.	🐑 1 Java Applet Alt+Shift+X, A :ip i
		Debug As	Þ	2 Java Application Alt+Shift+X, J Cla
		Validate		3 Mayloon Application
		Team	•	path
		Compare With		Run Configurations c'ha
		Compare with	•	[4:275] [INFO] [Convert] Project 'linpack' ha

Pic15 Run as Mayloon application



Select Java Application	
Select type (? = any character, * = any String, TZ = TimeZone):	-
**	
Matching items:	
CompressedTrieEn - org.feit.findword.trie Start - android.core TrieDictBuilder - org.feit.findword.trie	
android.core	OK Cancel

Pic16 Select Java application to run as MayLoon application

2. Start your desktop Chrome browser with the "--allow-file-access-from-files" command line option, and open the generated start page. If a Chrome instance is already running without this option, please close that first. Please check the log messages in the Developer Console of Chrome in case the application does not launch successfully.





Pic17 Converted application running on Chrome

5.4 Reviewing and Editing JavaScript

When you convert Android project to HTML5 project with MPT, it will generate JavaScript files in MayLoon specific syntax that general JavaScript editors do not recognize. MayLoon JavaScript editor helps application developers examine and edit the generated JavaScript files.

1. Open generated JavaScript files.

If you want to see the corresponding JavaScript file generated from a java file, Right click the java file and click "Edit *.js" from the popup menu.

MayLoon User Manual



Project Explorer 23	E	<u>, z (</u>
▲ ₩ linpack ▲ ₩ src		
a 🖶 android.c	ore	
⊳ 🗾 Star	New	
⊳ ∰ com.in	Open Type Hierarchy	F4
S gen Android Du	Show In	Al+, Chif+, M
Mayloon 0	Onen .	AILT SIIILT V
😕 assets	Open With	13
⊳ 🗁 bin	Open with	
external	Сору	Ctrl+C
🗁 mayloon_t	Copy Qualified Name	
	Paste	Ctrl+V
AndroidMi	K Delete	Delete
🔚 archive.zip	8_ Remove from Context	Ctrl+Alt+Shift+Down
linpack.htr	Build Path	
mayloon.b	Source	Alt+Shift+S
proguard.c	Refactor	Alt+Shift+T
È	import	
E	🔄 Export	
2	Refresh	F5
• Outline 😚 🔳	References	
n outline is not availa	Declarations	
	Edit Converted *.js	
	Run As	
	Debug As	
	Profile As	
	Validate	
	Edit *.js	N
	Team	4
	Compare With	
	Replace With	
	Restore from Local History	
	Properties	Alt+Enter
	1.500 5.50 202 5.50	

Pic18 Open corresponding JavaScript file

When you open a JavaScript file in an editor, if you want to navigate back to its corresponding java file, right click on the editor and click "Edit *.java" from the popup menu.



Clazz.declarePackage ("c c\$ = Clazz.decorateAsCla this.btnl = null:	om	<pre>intel.linpack"); (function () {</pre>	
<pre>this.bth1 = hull; this.mTextView = null; Clazz.instantialize (th }, com.intel.linpack, " Clazz.defineMethod (c\$, function (savedInstance Clazz.superCall (this, setContentView (2130903) bth1 = findViewById (21 mTextView = findViewById (21 clazz.supCaller); c\$.\$LinpackActivity\$642 Clazz.pu\$h (); c\$ = Clazz.declareAnony Clazz.defineMethod (c\$, function (arg0) {</pre>		Undo Revert File Save Open With Show In Cut Copy Paste Shift Right Shift Left Add to Snippets	Ctrl+Z Ctrl+S Alt+Shift+W Ctrl+X Ctrl+C Ctrl+V
var result = com.intel. mTextView setText (resu		Edit *.java	N
<pre>}, "View"); c\$ = Clazz.p0p (); }; //@ sourceURL=LinpackAc</pre>		Run As Debug As Profile As Validate Team Compare With Replace With Preferences	<i>λ</i> 3
		B () () ()	01 AF 0170 D

Pic19 Open corresponding Java file

2. Syntax Highlighting

When you open a JavaScript file in JavaScript editor, it will highlight its syntax words with different colors.



```
💰 Start.js 🔀
  Clazz.declarePackage ("android.core");
  c$ = Clazz.declareType (android.core, "Start");
  c$.main = Clazz.defineMethod (c$, "main",
 function (args) {
  var mainThread = ((Clazz.isClassDefined ("android.core.Start$318") ? 0 : and
  mainThread.start ();
  }. "~A"):
 c$.$Start$318$ = function () {
  Clazz.pu$h ();
  c$ = Clazz.declareAnonymous (android.core, "Start$318");
  Clazz.defineMethod (c$, "run",
  function () {
                   T
  try {
  Class.forName ("android.test.suitebuilder.annotation.HasClassAnnotation");
 } catch (e) {
 if (Clazz.exceptionOf (e, ClassNotFoundException)) {
  } else {
  throw e;
 3
 try {
 Class.forName ("android.test.suitebuilder.annotation.HasMethodAnnotation");
```

Pic20 Syntax Highlighting

When you double click one word or move cursor to one word, the editor will highlight the word selected and the same word in the document.

```
🕏 LinpackActivity.js 🖾
  Clazz.declarePackage ("com.intel.linpack");
  c$ = Clazz.decorateAsClass (function () {
  this.btn1 = null;
this.mTextView = null;
  Clazz.instantialize (this, arguments);
  }, com.intel.linpack, "LinpackActivity");
  Clazz.defineMethod (c$, "onCreate",
 function (savedInstanceState) {
 Clazz.superCall (this, com.intel.linpack.LinpackActivity, "onCreate", [savedInstanceSta
  setContentView (2130903040);
  btn1 = findViewById (2131034112);
  mTextView = findViewById (2131034113);
 btn1.setOnClickListener (((Clazz.isClassDefined ("com.intel.linpack.LinpackActivity$642
 }, "Bundle");
  c$.$LinpackActivity$642$ = function () {
 Clazz.pu$h ();
  c$ = Clazz.declareAnonymous (com.intel.linpack, "LinpackActivity$642");
  Clazz.defineMethod (c$, "onClick",
 function (arg0) {
  var result = com.intel.linpack.linpack.main ();
  mTextView.setText (result);
  }, "View");
 c$ = Clazz.p0p ();
 };
  //@ sourceURL=LinpackActivity.js
```

Pic21 Double click one word

3. Content Outline

When you open JavaScript editor, it will show the structure of the JavaScript file in the outline view. If a node in the content outline is selected then the cursor will move to the corresponding section in the text







Pic22 Content Outline

5.5 Debuging Application in Browser

If you fail to launch your converted application in browser, you can follow the below steps to identify the problem:

1. Take Chrome / Chromium browser as an example, press F12 to invoke the "Developer Tools" view, and click "Console" tab as below to check the console message, you can filter it with "Errors" at the bottom of the "Console" tab:



Banesk basewas ohner in Sorge in an instantion. Mainketivity being finished, but not in LRU list but in the sorget in the sorget in an instantion. Mainketivity being finished, but not in LRU list but the sorget in the sorg		
Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Contre Contro Control Control Control Control Co		
ActivityStack>>>> Activity Com.example.animation.MainActivity being finished, but not in LRU list bin/framework/android/util/ice.is: Activity: in startActivity/forBault bin/framework/android/util/ice.is: Activity: in startActivity/BayWait bin/framework/android/util/ice.is: PackageHonger: resolveIntent! bin/framework/android/util/ice.is: PackageHonger: resolveIntent? PackageHonger: resolveIntent? PackageRonger: resolveInten	🤮 Elements 🧓 Resources 🔞 Network 🎬 Scripts 🕂 Timeline 🕜 Profiles 🖳 Audits 📴 Console	Q Search Console
Activity: in startActivity/orResult bin/framework/android/utiliue.is: PackageInages: resolvEntent1 bin/framework/android/utiliue.is: PackageInages: resolvEntent2 bin/framework/android/utiliue.is: PackageInages: resolvEntent2 bin/framework/android/utiliue.is: PackageInages: resolvEntent2 bin/framework/android/utiliue.is: ActivityStack>>: startActivityMayWait2 bin/framework/android/utiliue.is: ActivityStack>>: startActivityMayWait2 bin/framework/android/utiliue.is: ActivityStack>>: startActivityMayWait2 bin/framework/android/utiliue.is: ActivityStack>>: startActivityMayWait2 bin/framework/android/utiliue.is: ActivityStack>>: startActivity Stack>>: startActivity bin/framework/android/utiliue.is: ActivityStack>>: startActivity is componet: com.example.android.patharc.arcsactivity bin/framework/android/utiliue.is: ActivityStack>>: startActivity is componet: com.example.android.patharc.arcsactivity bin/framework/android/utiliue.is: Activity: activit	ActivityStack>>>: Activity com.example.animation.MainActivity being finished, but not in LRU list	bin/framework/android/util/Log.js:99
ActivityStack>>>> startActivityMayNeit bit/framework/Android/util/Loc.Siz PackageManger: resolveIntent1 bit/framework/Android/util/Loc.Siz PackageManger: resolveIntent2 bit/framework/Android/util/Loc.Siz ActivityStack>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	Activity: in startActivityForResult	bin/framework/android/util/Log.js:95
<pre>packageInanger: resolveIntent2 PackageInanger: resolveInt</pre>	ActivityStack>>>: startActivityMayWait	bin/framework/android/util/Log.js:99
<pre>packageHanger: resolveIntent2</pre> bin/frameworK/android/util/Loc.fis: ActivityStack>>>: intent_getComponent: com.example.android.patharc, alnfo.name: com.example.android.patharc.ArcsActivity ActivityStack>>>: intent_getComponent: com.example.android.patharc, alnfo.name: com.example.android.patharc.ArcsActivity bin/frameworK/android/util/Loc.fis: ActivityStack>>>: intent_getComponent: com.example.android.patharc.ArcsActivity bin/frameworK/android/util/Loc.fis: ActivityStack>>>: intent_getComponent: com.example.android.patharc.ArcsActivity bin/frameworK/android/util/Loc.fis: ActivityStack>>>: intent_getComponent: com.example.android.patharc.ArcsActivity bin/frameworK/android/util/Loc.fis: ActivityStack>>>: intent_getComponent: com.example.android.patharc.ArcsActivity component: com.example.android.util/Loc.fis: ActivityStack>>>: intent_getComponent: com.example.android.patharc.ArcsActivity context isses: com.example.android.patharc/com.example.android.patharc.ArcsActivity context isses: com.example.android.util/Loc.fis: Context isses: com.example.android.util/Loc.fis: Context isses: com.example.android.patharc.ArcsActivity Context isses: com.example.android.util/Loc.fis: Context isses: com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity Context isses: com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.com.example.android.patharc.ArcsActivity.c	PackageManager: resolveIntent1	bin/framework/android/util/Log.js:99
PackageNanger: :1 bin/framework/android/util/Los.1s: ActivityStack>>>: star6.applicationTwfo.packageName: com.example.android.patharc.ArcsActivity bin/framework/android/util/Los.1s: ActivityStack>>>: star6.applicationTwfo.packageName: com.example.android.patharc.ArcsActivity bin/framework/android/util/Los.1s: ActivityStack>>>: attrivityStack>>>: star6.applicationTwfo.packageName: com.example.android.patharc.ArcsActivity bin/framework/android/util/Los.1s: ActivityStack>>>: attrivityStack>>>: attrivityStack bin/framework/android/util/Los.1s: ActivityStack>>>: attrivityStack bin/framework/android/util/Los.1s: ActivityStack bin/framework/android/util/Los.1s: Context: Lontext is:18 bin/framework/android/util/Los.1s: Context: Lontext is:19 bin/framework/android/util/Los.1s: Context: Lontext is:10 bin/framework/android/util/Los.1s: Context: Lontext is:20 bin/framework/android/util/Los.1s: Context: Context is:20 bin/framework/android/util/Los.1s: Context: Context is:21 bin/framework/android/util/Los.1s: Context: LongLicationContentResolver create 121 bin/framework/android/util/Los.1s: Context: LongLicationContentResolver: NoteLicate bin/framework/android/util/Los.1s: LayoutInflater: imflatesolver: Indexeover	PackageManager: resolveIntent2	bin/framework/android/util/Log.js:9
ActivityStack>>>: startActivityWait2 bin/framework/android/util/Loc.is: ActivityStack>>>: infent.getComponent: com.example.android.patharc, AncsActivity bin/framework/android/util/Loc.is: ActivityStack>>>: intent.getComponent: com.example.android.patharc,/com.example.android.patharc, AncsActivity bin/framework/android/util/Loc.is: ActivityStack>>>: AttivityStack>>>: AttivityStack>>>: Sending result to com.intel.jsdroid.home.Home (index 0) bin/framework/android/util/Loc.is: ActivityStack>>>: Nt >0; Nt: 4 bin/framework/android/util/Loc.is: ActivityStack>>>: Nt >0; Nt: 4 bin/framework/android/util/Loc.is: Activity: Activity's CanonicalImae: com.intel.jsdroid.home.Home, onPause bin/framework/android/util/Loc.is: Context: ApplicationContentResolver create 118 bin/framework/android/util/Loc.is: Context: IdeplicationContentResolver:NOT NULL bin/framework/android/util/Loc.is: Context: Scill bin/framework/android/util/Loc.is: Context: Context is:18 bin/framework/android/util/Loc.is: Context: Context is:20 bin/framework/android/util/Loc.is: Context: ApplicationContentResolver:Not NULL bin/framework/android/util/Loc.is: Context: ApplicationContentResolver:Not NULL bin/framework/android/util/Loc.is: Context: ApplicationContentResolver:Not NULL bin/framework/android/util/Loc.is:	PackageManager: :1	bin/framework/android/util/Log.js:99
ActivityStack>>>> info.applicationInfo.packageMane: com.example.android.patharc, ArcsActivity bin/framework/android/util/Loc.is: ActivityStack>>>> into: applicationInfo.packageMane: com.example.android.patharc, ArcsActivity bin/framework/android/util/Loc.is: ActivityStack>>>> into: applicationInfo.packageMane: com.example.android.patharc, ArcsActivity bin/framework/android/util/Loc.is: ActivityStack>>>> into: applicationInfo.packageMane: com.example.android.patharc, ArcsActivity ActivityStack>>>> into: applicationInfo.packageMane: com.example.android.patharc, ArcsActivity, concerts applicationInfo.packageMane: com.example.android.patharc, ArcsActivity, onCreate Activity: Activity: Standords/Info.famework/android/Util/Loc.is: Din/framework/android/Util/Loc.is: Context is:18 Context is:20	ActivityStack>>>: startActivityMayWait2	bin/framework/android/util/Log.js:99
ActivityStack>>>: Intent.getComponent: com.example.android.patharc/.om.example.android.patharc/.ancsActivity bin/framework/android/util/Los.is: ActivityStack>>>: ActivityStack>>>: ActivityStack>>>: ActivityStack>>>: ActivityStack>>>: ActivityStack>>>: ActivityStack>>>: Nt > 0; Mt: 4 bin/framework/android/util/Los.is: ActivityStack>>: Nt > 0; Mt: 4 bin/framework/android/util/Los.is: bin/framework/android/util/Los.is: Activity: Activity's CanonicalMame: com.intel.jdroid.home.Home, onPause bin/framework/android/util/Los.is: Context: Context is:18 bin/framework/android/util/Los.is: Context: Context is:19 bin/framework/android/util/Los.is: Context: Context is:20 bin/framework/android/util/Los.is: Context: Context is:21 bin/framework/android/util/Los.is: Context: Context is:20 bin/framework/android/util/Los.is: Context: Context is:21 bin/framework/android/util/Los.is: Context: Context is:22 bin/framework/android/util/Los.is: Context: Context is:23 bin/framework/android/util/Los.is: Context: ApplicationContentResolver (NOT NULL bin/framework/android/util/Los.is: Context: ApplicationContentResolver (NOT NULL bin/framework/android/util/Los.is: LayoutInflater: createView:FrameLayout Donel: UID: 148, ID: 16908290 bin/framework/android/util/Los.is: LayoutIn	ActivityStack>>>: aInfo.applicationInfo.packageName: com.example.android.patharc, aInfo.name: com.example.android.patharc.ArcsActivity	bin/framework/android/util/Log.js:99
ActivityStack>>>: ActivityStack>>>: ActivityStack>>>: ActivityStack>>>: ActivityStack>>>: Mix 4 bin/framework/android/util/Log.is: ActivityStack>>>: Mix 4 bin/framework/android/util/Log.is: Activity: Activity's CanonicalName: com.intel.jsdroid.home.Home (index 0) bin/framework/android/util/Log.is: Context: Sills bin/framework/android/util/Log.is: Context: ApplicationContentResolver create 118 bin/framework/android/util/Log.is: Context: IApplicationContentResolver create 118 bin/framework/android/util/Log.is: Context: Context is:19 bin/framework/android/util/Log.is: Context: Context is:20 bin/framework/android/util/Log.is: Context: ApplicationContentResolver create 121 bin/framework/android/util/Log.is: Context: ApplicationContentResolver create 122 bin/framework/android/util/Log.is: Context: ApplicationContentResolver NUL bin/framework/android/util/Log.is: Context: ApplicationContentResolver NUL bin/framework/android/util/Log.is: LayoutFriater: creatView: Graedules: cnextr: ApplicationContentResolver NUL LayoutFriater: creatView: success bin/framework/android/util/Log.is: LayoutFriater: creatView: success bin/framework/android/util/Log.is: LayoutFriater: creatView: success bin/framework/android/util/Log.is:	ActivityStack>>>: intent.getComponent: com.example.android.patharc/com.example.android.patharc.ArcsActivity	bin/framework/android/util/Log.js:9
Activity: Activ	ActivityStack>>>: ActivityStack>>>Sending result to com.intel.jsdroid.home.Home (index 0)	bin/framework/android/util/Log.js:99
Activity: Activity: Sanotcallame: com.intel.jsdroid.home.Home, onPause bin/framework/android/util/Log.fs: Context: Context is:18 bin/framework/android/util/Log.fs: Context: ApplicationContentResolver:NOT NULL bin/framework/android/util/Log.fs: Context: IsplicationContentResolver:NOT NULL bin/framework/android/util/Log.fs: Context: Context is:18 bin/framework/android/util/Log.fs: Context: Context is:20 bin/framework/android/util/Log.fs: Context: Context is:21 bin/framework/android/util/Log.fs: Context: ApplicationContentResolver reste 121 bin/framework/android/util/Log.fs: Context: ApplicationContentResolver reste 121 bin/framework/android/util/Log.fs: Context: ApplicationContentResolver reste 121 bin/framework/android/util/Log.fs: Activity: Activity: Sanotcallame: com.example.android.patharc.ArcsActivity, onCreate bin/framework/android/util/Log.fs: LayoutInFlater: "metawork/android/util/Log.fs: bin/framework/android/util/Log.fs: LayoutInFlater: inflate>/screatVies bin/framework/android/util/Log.fs: LayoutInFlater: inflate>/screatVies bin/framework/android/util/Log.fs: LayoutInFlater: inflate>/screatVies bin/framework/android/util/Log.fs: LayoutInFlater: inflate>/screatVies bin/framework/android/util/Log.fs:	ActivityStack>>>: NH > 0, NH: 4	bin/framework/android/util/Log.js:99
Context: Din/framework/android/util/Loc.is: Context: ApplicationContentResolver:NOT NUL Din/framework/android/util/Loc.is: Context: ApplicationContentResolver:NOT NUL Din/framework/android/util/Loc.is: Context: InplicationContentResolver:NOT NUL Din/framework/android/util/Loc.is: Context: Context: Din/framework/android/util/Loc.is: Context: Context: Din/framework/android/util/Loc.is: Context: Context: Din/framework/android/util/Loc.is: Context: ApplicationContentResolver create 121 Din/framework/android/util/Loc.is: Context: InplicationContentResolver: Din/framework/android/util/Loc.is: Context: InplicationContentResolver: Din/framework/android/util/Loc.is: Context: InplicationContentResolver: Din/framework/android/util/Loc.is: LayoutInFlater: Inflatesolver: Din/framework/android/util/Loc.is: LayoutInFlater: Inflatesolver: Din/framework/android/util/Loc.is: LayoutInFlater: Inflatesolver: Din/framework/android/util/Loc.is: LayoutInFlater: Inflatesolver: Din/framework/android/util/Loc.is: LayoutInFlater: Inflatesolvere: Din	Activity: Activity's CanonicalName: com.intel.jsdroid.home.Home, onPause	bin/framework/android/util/Log.js:95
Context: ApplicationContentResolver create 118 bin/framework/android/util/Los.is: Context: IApplicationContentResolver:NOT NULL bin/framework/android/util/Los.is: Context: Context is:19 bin/framework/android/util/Los.is: Context: Context is:20 bin/framework/android/util/Los.is: Context: Context is:21 bin/framework/android/util/Los.is: Context: ApplicationContentResolver:NOT NULL bin/framework/android/util/Los.is: Context: ApplicationContentResolver:NOT NULL bin/framework/android/util/Los.is: Activity: Activity's CanonicalName: com.example.android.audget. bin/framework/android/util/Los.is: LayoutInitate: createView: irameLayout Conel UID: 148, ID: 16908290 bin/framework/android/util/Los.is: LayoutInitate:: inflate>: carested: bin/framework/android/util/Los.is: LayoutInitate:: inflate>: carest	Context: Context is:18	bin/framework/android/util/Log.is:95
Context: iPplicationContentResolverINOT NULL bin/framework/android/util/Los.is: Context: Context is:19 bin/framework/android/util/Los.is: Context: Context is:20 bin/framework/android/util/Los.is: Context: Context is:21 bin/framework/android/util/Los.is: Context: Context is:20 bin/framework/android/util/Los.is: Context: IsplicationContentResolver create 121 bin/framework/android/util/Los.is: Context: ApplicationContentResolver NOT NULL bin/framework/android/util/Los.is: Activity: Activity: ScanoicalName: con.example.android.patharc.ArcsActivity, onCreate bin/framework/android/util/Los.is: LayoutInFlater: createView: Inframework/android/util/Los.is: bin/framework/android/util/Los.is: LayoutInFlater: inframework/android.patharc.ArcsActivity, onStart bin/framework/android/util/Los.is:	Context: ApplicationContentResolver create 118	bin/framework/android/util/Log.js:95
Context: 1:01 Din/framework/android/util/Los.is: Context: Context 1:s:19 Din/framework/android/util/Los.is: Context: Context 1:s:20 Din/framework/android/util/Los.is: Context: ApplicationContentResolver create 121 Din/framework/android/util/Los.is: Context: Context 1:s:21 Din/framework/android/util/Los.is: Context: ApplicationContentResolver create 121 Din/framework/android/util/Los.is: Activity: Intracollame: cone.example.android.patharc.ArcsActivity, onCreate Din/framework/android/util/Los.is: Activity: Conto:: Inflato:: FrameLayout:android.widget. Din/framework/android/util/Los.is: LayoutInflater: Creating ulew: FrameLayout:android.widget. Din/framework/android/util/Los.is: LayoutInflater: Inflato:: Inflato:: Creating ulew: FrameLayout:Bandroid.widget. Din/framework/android/util/Los.is: LayoutInflater: Inflato:: Inflato:: Creating ulew: FrameLayout:Bandroid.patharc.ArcsActivity, onStart Din/framework/android/util/Los.is: Activity: Activity: Activity: ScannicalName: cone.example.android.patharc.ArcsActivity, onResume Din/framework/android/util/Los.is: ViewRoot: ViewRoot: Din/framework/android/util/Los.is: ViewRoot: Din/framework/android/util/Los.is: Din/framework/android/util/Los.is: ViewRoot: Vie	Context: ApplicationContentResolver:NOT NULL	bin/framework/android/util/Log.is:9
Context: So:28 bin/framework/android/util/Los.is: Context: Context: So:28 bin/framework/android/util/Los.is: Context: ApplicationContentResolver create 121 bin/framework/android/util/Los.is: Context: ApplicationContentResolver create 121 bin/framework/android/util/Los.is: Context: ApplicationContentResolver NOT NULL bin/framework/android/util/Los.is: ActivityThread: appContext is created: bin/framework/android/util/Los.is: Activity: Activity's CannoicalName: com.example.android.patherc.ArcsActivity, onCreate bin/framework/android/util/Los.is: LayoutInFlater: createView uscess bin/framework/android/util/Los.is: LayoutInFlater: inflate>: com.example.android.patherc.ArcsActivity, onStart bin/framework/android/util/Los.is: LayoutInFlater: inflate>: com.example.android.patherc.ArcsActivity, onStart bin/framework/android/util/Los.is: LayoutInFlater: inflate>: com.example.android.patherc.ArcsActivity, onStart bin/framework/android/util/Los.is: LayoutInFlater: inflate>: com.example.android.patherc.ArcsActivity, onResume bin/framework/android/util/Los.is: LayoutInFlater: inflate>: com.example.android.patherc.ArcsActivity, onResume bin/framework/android/util/Los.is: ViewRoot: ViewNisbilityChanged: 0 bin/framework/android/util/Los.is: ViewRoot: ViewVisbilityChanged: 0 bin/framework/android/util/Los	Context: Context is:19	bin/framework/android/util/Log.is:9
Context: Context is:21 bin/framework/android/util/Log.is: Context: ApplicationContentResolver create 121 bin/framework/android/util/Log.is: Context: ApplicationContentResolver create 121 bin/framework/android/util/Log.is: Context: ApplicationContentResolver create 121 bin/framework/android/util/Log.is: Context: ApplicationContentResolver NOT NUL bin/framework/android/util/Log.is: Activity: Activity's ContocallAme: come.example.android.patharc.ArcsActivity, onCreate bin/framework/android/util/Log.is: LayoutInflater: createView:FrameLayout Bone! UIID: 148, ID: 16908290 bin/framework/android/util/Log.is: LayoutInflater: inflate>>>createView:GrameWork/android.patharc.ArcsActivity, onStart bin/framework/android/util/Log.is: Activity: Activity's CannoicalName: com.example.android.patharc.ArcsActivity, onResume bin/framework/android/util/Log.is: Activity: Activity's CannoicalName: com.example.android.patharc.ArcsActivity, onResume bin/framework/android/util/Log.is: Activity: Activity's CannoicalName: com.example.android.patharc.ArcsActivity, onResume bin/framework/android/util/Log.is: ViewRoot: viewVisibility: Activity's CannoicalName: com.example.android.patharc.ArcsActivity, onResume bin/framework/android/util/Log.is: ViewRoot: viewVisibility: Activity is CannoicalName: com.example.android.patharc.ArcsActivity, onResume bin/framework/android/util/Log.is: <	Context: Context is:20	bin/framework/android/util/Log.is:95
Context: ApplicationContentResolver create 121 bin/framework/android/util/Los.is: Context: ApplicationContentResolver create 121 bin/framework/android/util/Los.is: Context: IApplicationContentResolver create 121 bin/framework/android/util/Los.is: ActivityInexci: appCintext is created: 121 bin/framework/android/util/Los.is: Activity: Activity: CannicalName: com.example.android.petherc.ArcsActivity, onCreate bin/framework/android/util/Los.is: LayoutInFlater: creativies wicesis bin/framework/android/util/Los.is: LayoutInFlater: inflate-youresetVies wicesis bin/framework/android/util/Los.is: LayoutInFlater: inflate-youresetVies wicesis bin/framework/android/util/Los.is: LayoutInFlater: inflate-youresetVies wicesis bin/framework/android/util/Los.is: Activity: Activity's CannicalName: com.example.android.patharc.ArcsActivity, onStart bin/framework/android/util/Los.is: Activity: Activity's CannicalName: com.example.android.patharc.ArcsActivity, onResume bin/framework/android/util/Los.is: ViewRoot: itaching handlers bin/framework/android/util/Los.is: ViewRoot: itaching handlers bin/framework/android/util/Los.is: ViewRoot: itaching handlers bin/framework/android/util/Los.is: ViewRoot: itaching handlers bin/framework/android/util/Los.is: ViewRoot: itaching h	Context: Context is:21	bin/framework/android/util/Log.is:95
Context: !ApplicationContentResolver!NOT NULL bin/framework/android/util/Log.is: Activity: Activity's ConscialMame: come.example.android.patharc.ArcsActivity, onCreate bin/framework/android/util/Log.is: Activity: Activity's ConscialMame: come.example.android.patharc.ArcsActivity, onCreate bin/framework/android/util/Log.is: LayoutInFlater: creativity's ConscialMame: come.example.android.patharc.ArcsActivity, onCreate bin/framework/android/util/Log.is: LayoutInFlater: creativity's ConscialMame: come.example.android.patharc.ArcsActivity, onCreate bin/framework/android/util/Log.is: LayoutInFlater: inflates: scaleWise: ScaleWise bin/framework/android/util/Log.is: LayoutInFlater: inflates: for View: 148 bin/framework/android/util/Log.is: Activity: Activity's ConscialMame: come.example.android.patharc.ArcsActivity, onResume bin/framework/android/util/Log.is: Activity: Activity's ConscialMame: come.example.android.patharc.ArcsActivity, onResume bin/framework/android/util/Log.is: ViewRoot: XiewVisibility: Activity's ConscialMame: come.example.android.patharc.ArcsActivity, onResume bin/framework/android/util/Log.is: ViewRoot: XiewVisibility: Activity's ConscialMame: come.example.android.patharc.ArcsActivity, onResume bin/framework/android/util/Log.is: ViewRoot: XiewVisibility: Activity: Activity: Scial Conscience: Come.example.android.patharc.ArcsActivity, onResume bin/framework/android/util/Log.is: ViewRoot: Xiew	Context: ApplicationContentResolver create 121	bin/framework/android/util/log.is:9
ActivityThread: appContext is created:21 bin/framework/android/util/Log.is: ActivityThread: appContext is created:21 bin/framework/android/util/Log.is: Activity: Activity: ScannicalName: com.example.android.patharc.ArcsActivity, onCreate bin/framework/android/util/Log.is: LayoutInFlater: ************************************	Context: ApplicationContentResolverINOT NULL	hin/framework/android/util/log.is:9
Activity: Activity: StandialMame: com.example.android.patharc.ArcsActivity, onCreate LayoutInFlater: creativitew: FrameLayout:android.widget. LayoutInFlater: creativitew: FrameLayout Done! UIID: 148, ID: 16906290 LayoutInFlater: inrifate>>>creativitew: FrameLayout Done! UIID: 148, ID: 16906290 LayoutInFlater: inrifate>>>creativitew: Strameork/android/util/Los.is: LayoutInFlater: inrifate>>>creativitew: Strameork/android/util/Los.is: LayoutInFlater: inrifate>>>creativitew: 148 Activity: Activity's CannoicalMame: com.example.android.patharc.ArcsActivity, onStart Activity: Activity: ScannicalMame: com.example.android.patharc.ArcsActivity, onResume Din/framework/android/util/Los.is: ViewRoot: Xitaking handlers ViewRoot: Xita	ActivityThread: anDContext is created:21	hin/framework/android/util/log.is:99
LayoutInflater: createView:FrameLayout:android.widget. LayoutInflater: createView:FrameLayout:android.widget. LayoutInflater: createView:FrameLayout:android.widget. LayoutInflater: createView:FrameLayout:android.widget. LayoutInflater: inflate>>creating: createView:CreateView:CreateView: LayoutInflater: inflate>>createView:FrameLayout:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView:CreateView	Activity: Activity's Canonicallame: com example android patharc AccsActivity, opCreate	hin/framework/android/util/log_is-95
LayoutInitate: in/framework/android/util/Log.is: LayoutInitate: in/framework/android/util/Log.is: LayoutInitate: in/framework/android/util/Log.is: LayoutInitate: bin/framework/android/util/Log.is: LayoutInitate: bin/framework/android/util/Log.is: LayoutInitate: bin/framework/android/util/Log.is: LayoutInitate: bin/framework/android/util/Log.is: ViewRoot: kitvity: Kativity: Activity: Kativity: Activity: ViewRoot: kitvity: Kativity: Activity: Kativity:	avait of later: createlying Franciscust control widet	bin/framework/android/util/log_is:90
Layoutfritter: infirmmenck/android/util/Log.is: Layoutfritter: infirmmenck/android/util/Log.is: Layoutfritter: infirmmenck/android/util/Log.is: Layoutfritter: infirmmenck/android/util/Log.is: Activity: Activity:	Layouth fater, statistic for a fing view. Frame avoid formal UITD: 148 ID: 16008200	bin/framework/android/util/log_is:00
Layoutinities: in/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initiator/initinininitiator/initiator/initiator/initiator/init	Layouthinite: inflata://inflata//instati/instar/instar/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati/instati	hin/framework/android/util/log_is:00
Laboratization: Display the function of the func	Layouth filter - Initiate//references	bin/framework/android/util/log_is:0
Activity: Activity: ScannicalName: com.example.android.patharc.ArcsActivity, onResume Din/framework/android/util/Log.is: ViewRoot: Attaching handlers Din/framework/android/util/Log.is: ViewRoot: Attaching handlers Din/framework/android/util/Log.is: ViewRoot: Attaching handlers Din/framework/android/util/Log.is: Din/framework/android/util/Log.is: Din/framework/android/Util/Log.is:	cujotantaceri an tantace, to talet ato	bin/framework/android/util/log.js.95
ViewRoot: Attaching handlers Din/framework/android/viii/log.is. ViewRoot: viewVisibilityChanged: 0 Din/framework/android/viii/log.is. Home>>>:	Artivity: Artivity's CanonicalName: com example android, nature: ArsArtivity, onResume	bin/framework/android/util/Log_is-05
ViewRoot: viewVisibilityChanged: 0 Home>>>:Application launch time is:211 DiscremeNorK/android/util/Log.is: DiscremeNorK/android/util/Log.is: DiscremeNorK/android/util/Log.is:	ViauRont: Attaching bandlers	hin/framework/android/util/Log_is-0
Home>>>:Application launch time is:211	ViewDort receiving instances	bin/framework/android/util/Log.is.3
	Hempshiri Henrichter Hempshiri eine ist 211	bin/framework/android/util/Log.is.9
2) 2 Q Q <top frame=""> + M (Frors) Warnings Logs @11</top>	Index Ppyaaataan aminin take askaa	gain, in direction R/ Bildi Old/ dc11/ LOE. 13:35
		8 11

Pic23 Chrome console

 As the errors in "Console" are printed out when the errors already occur, if you want to pause at the stack when errors occur you can switch to "Sources" (It can be "Scripts" varying from Chromium versions) tab and enable "Pause all exceptions" at the bottom. Then press F5 to refresh the page to reload the application, and it will pause at the exceptions.

Other than relying on runtime exceptions, MayLoon also provides the force breakpoints for the developer. It will force the debugger to break at that line if you have opened the Chrome Inspector, regardless there is exception or not. You can insert the breakpoint by using either of the below functions in Java code:

DebugUtils.DebugBreakPoint(); // for unconditional breakpoint

DebugUtils.DebugBreakPoint(boolean condition); // for conditional breakpoint

If you want to force break at JavaScript level, you can use below code:

debugger; // break at JavaScript level



Elements Resources Network Sources Timeline Profiles Audits Console						0
D android.core.Start.html x	11	11 2	÷	î 🚧		
48 /*forward : true.*/		▶ Watch	Expressi	ons	+	C
<pre>49 mode : "dailybuild",</pre>		▼ Call St	ack			
50 onload : function () {		▼ Scope	Variable			
51 Var j25085e = Window[/j25.llp].j25085e; 52 Claridade narkaedlasenath ("suga" j650ace true).		T Decale	- inte			
Sa Clazicoder, setPrimarvfolder ("bin/framework"):		* DIedk	Joints			
54 ClazzLoader.packageClasspath (["com.intel.jsdroid.com.dujin.amail", "com.imps.tabletennis.tranning", "java.net", "com.intel.jsdroid.weather", "android.graph	ic:		No Bri	akpoints:		
55 ClazzLoader.loadClass ("android.core.Start", function () {		► DOM	Breakpoi	ats		
50 anuroid.core.scart.main([]); 57 }).		► XHR P	reakpoin	ts		+
58		▶ Event	listener	Breaknoin	ts	
59 };		▶ Work	arc	reaupon		
60		PHOING	13			-
<pre>62 // Following script will make J2SLib compatible with Java2Script addon function generateScriptCallback () { 64 return function () { 75 war s = this:readyState; 76 if (s=readyState; 77 this.onceadystatechange = null; 78 this.onceadystatechange = null; 79 this.onceadystatechange = null; 71 this.onload = null; 72 } 73 }; 74 following script will make SCRIPT"); 75 function loadJ2SLibZJS (path, cb) { 76 var sxr = document.createElement ("SCRIPT"); 77 sxr.scr = path; 78 sxr.type = "text/javascript"; 79 if (typeof sxr.tpl = "undefind=") { 79 sxr.scr = path; 79 var t = "aneadystatechange"; 79 var str = document.createIlback (); 79 var str = document.scriptCallback (); 79 var t = "aneadystatechange"; 70 var str = document.scriptCallback (); 71 var str = document.scriptCallback (); 72 var str = document.scriptCallback ; 73 j; 74 j; 75 var t = "aneadystatechange"; 75 var str = document.scriptCallback ; 75 va</pre>						
91 Click to Pause on uncaught exceptions.	<u>,</u>					-
					O 24	쮸

Pic24 Chrome console debugging

2. You can click "Pretty Print" to make the JavaScript code in the panel more readable (with proper indent).



Pic25 Chrome console format

3. You can check "Call Stack", "Watch Expressions", "Scope Variables" and also "Step into the function" in the Debug Area at the right of the source code panel. You can also search methods in the "Search Scripts" toolbar.



						Q Search Scripts	0
111	IÞ-	3	÷	1	v/a	0	Pauser
*	▼W	atch I	Expre	ssior	IS	+	C
						No Watch Expressions	
	W Ca	all Sta	ck				
	de	legatir	19				
	CI	azz.loa	d.Cla	zz.defi	neMethod	Sfz	:56
	(3	nonym	ous fui	nction)			:62
	CI	azz.de	fineMe	ethod.v	iewGroup	bin/framework/android/view/View.js:1	881
	CI	azz.ov	erridel	lethod	.children	bin/framework/android/view/ViewGroup.js:	821
	CI	azz.de	fineMe	ethod.r	nore	bin/framework/android/view/ViewGroup.js:	489
	CI	azz.sae	m			/C:/Java/eclipse-java-juno-win32-x86_64/eclipse/plugir	1
	de	elegatir	19			/C:/Java/eclipse-java-juno-win32-x86_64/eclipse/plugir	1
	CI	azz.ov	erridel	lethod	l.children	bin/framework/android/view/ViewGroup.js:	819
	CI	azz.de	fineMe	ethod.r	nore	bin/framework/android/view/ViewGroup.js:	489
	CI	azz.sae	em.			/C:/Java/eclipse-java-juno-win32-x86_64/eclipse/plugin	1
	de	legatir	19			/C:/Java/eclipse-java-juno-win32-x86_64/eclipse/plugir	1
	CI	azz.de	fineMe	thod.	iewGroup	bin/framework/android/view/View.js:14	885
	CI	azz.de	fineMe	athod.\$	6fz	bin/framework/android/view/ViewRoot.js:	752
	CI	azz.de	fineMe	athod.	Sfz	bin/framework/android/view/ViewRoot.js:	561
	(3	nonym	ous fui	nction)		bin/framework/android/view/ViewRoot.js:	778
	(a	nonym	ous fu	nction)		bin/framework/android/os/Handler.js	:22
	CI	azz.de	fineMe	ethod.\$	Sfz	bin/framework/android/os/MessageQueue.js:	140
	CI	azz.de	fineMe	ethod.r	nsg	bin/framework/android/os/MessageQueue.js:	112
	CI	azz.de	fineMe	thod.s	ent	bin/framework/android/os/Handler.js:	156
	(a	nonym	ous fui	nction)		bin/framework/android/os/Handler.js:	143
-	▼ Se	cope \	/arial	bles			
	T Lo	ocal					

Pic26 Stack info

4. Sometimes we can't step into the function directly. We need to reach the function by going through some internal low-level function calls (This is depending on the compiler's implementation).



Pic27 Application debug

5.6 Packaging for Tizen

If you want to package the converted application for Tizen, please follow the steps below after converting the project to MayLoon:



- 1. Convert and build the Android project as Mayloon Application
- 2. Right click project and then select Export...
- 3. Expand Mayloon and select Export Mayloon Application and click Next as follows

Export				
Select				ß
Select an export des	ination:			
Image: Second secon	1ayloon Applicatio elopment	<u>n</u>		
?	< Back	Next >	Finish	Cancel

Pic28 Export application

4. On this page of export wizard, select the project you want to export and click Finish. By default the Tizen project will be exported to mayLoon_bin folder under your project folder. You can also check "Enable JavaScript compression" to reduce the size of all JavaScript files during exporting if you want.



•	Export MayLoon Application -	
Select Project to	Export	
Will export the sele	ected project to the destination directory.	
Project:	Amail	Browse
Export destination:	C:/Users/longxian/workspace_1/Amail/mayloon_bin	Browse
 Enable JavaScrip 	t compression	
?	< Back Next > Finish	Cancel

Pic29 Export wizard

- 5. Follow the page http://developer.tizen.org/downloads/sdk/installing-tizen-sdk to setup your Tizen development environment.
- 6. Open Tizen IDE and create a new Tizen web project



ile	Edit Navigate Search Project	Run	Window	Help			
	New		Alt+Shi	ft+N ▶	<u> </u>	Tizen Web Project	
	Open File				C2	Project	
	Close		Ct	rl+W	CĴ	Other	Ctrl+N
	Close All		Ctrl+Shi	ft+W			
	Save		C	trl+S	L .		
	Save As				L .		
	Save All		Ctrl+Sh	ift+S	L .		
	Revert						
	Move				L .		
1	Rename			F2	L .		
1	Refresh			F5	L .		
	Convert Line Delimiters To			•			
5	Print		C	trl+P	L .		
	Switch Workspace			×	L .		
	Restart				b r		
	Import				H		
3	Export						
	Properties		Alt+	Enter	F		
	1 config.xml [amail]						
	2 config.xml [hairstyle]				1		
	3 config.xml [test]				1		
	4 config.xml [com.intel.jsdroid.calcu	1				Problems 🕅	N 🗉 o

Pic30 Create Tizen web project

7. If you are working with Tizen IDE **2.0**, do not use default location, and set Location with content exported in step 5.



Pic31 Set project location



Otherwise if you are using Tizen IDE **2.1**, please create a new Tizen web project and import your exported MayLoon application to this newly created project as following shows. Please confirm if prompted to overwrite existing resources.

elect an import source:	
type filter text	
 General General Archive File Existing Projects into Workspace File System File System Preferences Android General C/C++ C/S General File System File System<th>E</th>	E

Pic32 Select import source



Import		2.10.04		
Import resources from the local file	e system.			
From directory: C:\Share\source\	mayloon-qa\Andr	oidAppSource\linpa	ck\mayloon_bi 👻	Browse
▷ ♥ 🧽 mayloon_bin		Com.inte	l.linpack.LinpackAc	tivity.html
Into folder: linpack	Deselect All			Browse
Options				
Overwrite existing resources wi	thout warning			
Create top-level folder Advanced >>				
•				
(7)	< Back	Nevt >	Finish	Cancel

Pic33 Import from File System

8. Open config.xml in created project; modify index.html in Content tab to the generated html file.

B Tizen Web - linpack/config.xml - Tizen IDE	State of Sta	In the set of the local sectors, where	Street Res
File Edit Navigate Search Project Run Window Help			
C1 ▼ □ 10 △ □ 10 ▼ 0 ▼ 0 ▼ 0 ▼ 0 ▼ 0 ▼ 0 □ □			
Project Explorer 🛛 🔄 😫 📔 🌄 🖓 🖓 🖓 🖓 🖓 🖓 🖓			
📂 linpack	Overview		
➡ JavaScript Resources			(
🔁 bin	General Infomation		Widget Content
	This section describes general information about this		The content of the widget is made up of the following
external	widget.		section(s):
🗁 inages	Identifier*	http://yourdomain/linpack	ô <u>Widget</u>
		100	Features
🥟 res_sys	Version	1.0.0	
🗁 WebContent	Name	linpack	
👩 config.xml	Content	linpack.html Browse	Access
📇 icon.png			Localization
index.html	Icon	icon.png Browse	Preference
Inpack.html			
	Launcher Icon	:	
Connection Explorer 🕱 🗧 🗖			Tizen Content
			The content of the tizen is made up of the following section(s):
Files			🔅 Tizen
	Overview Widget	Features Privileges Access Localization	Preferences Tizen Source

Pic34 Set Content html file



- 9. Right click the project and click Run As->Tizen Web Simulator Application to run the project in Tizen web simulator.
- 10. If you want to run the application on a Tizen device, you should build the imported project in Tizen IDE, and a Tizen application widget (.wgt file) will be generated.

Note: In order to speed up project build, please go to project Properties->Builders, then unselect Validator and JSLint Builder items. If you keep these builder options selected, the build process will be much longer. There might be compile errors in the project although they will not impact the generated package.



6 Supported APIs

Please refer to one of the MayloonAPISupport HTML documents to find the list of APIs supported by MayLoon alpha.



7 Unsupported Features

This chapter lists some of the features available from Android application framework while currently not available from MayLoon.

If some features are not supported, the corresponding API will not function as expected, but this does not necessarily mean that MayLoon will not be able to convert those Android applications.

• Google service

MayLoon has no access to Google service and APIs for now

• TTS

Text to speech APIs are not supported for now

• Notification

MayLoon currently doesn't have notification support.

• Security model

MayLoon does not support Android's security model. Instead, the resulting HTML5 applications work in web browsers and web runtimes and are governed by their respective security models. Therefore behavior with security impact in MayLoon could be different from Android.

• External storage

MayLoon cannot access external storage on the devices yet.

• Power management

MayLoon does not support Android power management API and schemes.

• GPS location provider

MayLoon currently doesn't support GPS location provider.

• Camera

MayLoon applications cannot use camera yet.

Wallpaper

The resulting HTML5 applications cannot operate on wallpaper.

AppWidget

• Predefined intents



Intents is mechanism that Android implements to support message delivery between Android application and system. MayLoon currently does not support any of the predefined Intents – for example placing phone call or launching web browser.

- Bluetooth operation
- Networking features
- Telephony features
- Vibrator
- File IO
- Sensors



8 Known Incompatibilities and Limitations

8.1 Eclipse plugin

On Windows PC, if the Eclipse installation and the application are on different disk drives (C: /, D :/), the resulting HTML5 application might not be able to run.

8.2 Incompatibility between Java and JavaScript

This section lists the known incompatibilities when converting with MayLoon.

Currently, the MayLoon porting tool does not prompt the developer that any of the incompatibilities is identified in the original Android application.

As MayLoon translates between 2 different programming languages, some differences are inevitable:

• **Function overload:** Function overloading is a usual feature in Object Oriented languages. MayLoon implements majority of function overloading capability. However, consider the following Java class:

```
public class MyClass {
    public static void method(int intParam){
        System.out.println("int");
    }
    public static void method(float floatParam){
        System.out.println("float");
    }
    public static void method(String strParam){
        System.out.println("String");
    }
    public static void main(String[] args) {
        method(0); // Call method(<u>int</u>)
method(1.2f); // Call method(float)
        method("Hello");
                             // Call method(String)
    }
}
```



One of the differences between JavaScript and Java is the numbering system. In JavaScript there is only one 'Number' type for all the numbers. Therefore, when converted to JavaScript, the application is not able to distinguish between

method(int intParam)

and

method(float floatParam).

When method(0) is invoked in the example, the program outputs "float" instead of "int". The application is able to distinguish between

method(String strParam)

from the other two though.

• **64-bit Bitwise Operators**: Bitwise operators are also translated from Java to JavaScript in a straightforward way. For example, bitwise AND

a & b

is translated into the same operation. In JavaScript, all numbers are stored as 64-bit (8-bytes) floating point numbers. However, JavaScript bitwise operators do not work on all 64 bits of the operands. Actually they only work on half of the bits - 32 bits. Therefore when the Android application performs bitwise operations on **long** integers, the behavior of the resulting HTML5 application might be different.

• **Multi-process and multi-thread**: Due to the single-threaded nature of JavaScript, MayLoon applications do not support multi-process or multi-thread. Threading behavior of MayLoon applications is undefined.



9 Upcoming Features

The following are some features we will add in upcoming releases:

- Content Provider
- Simple services