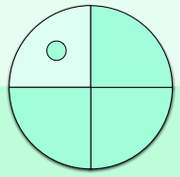
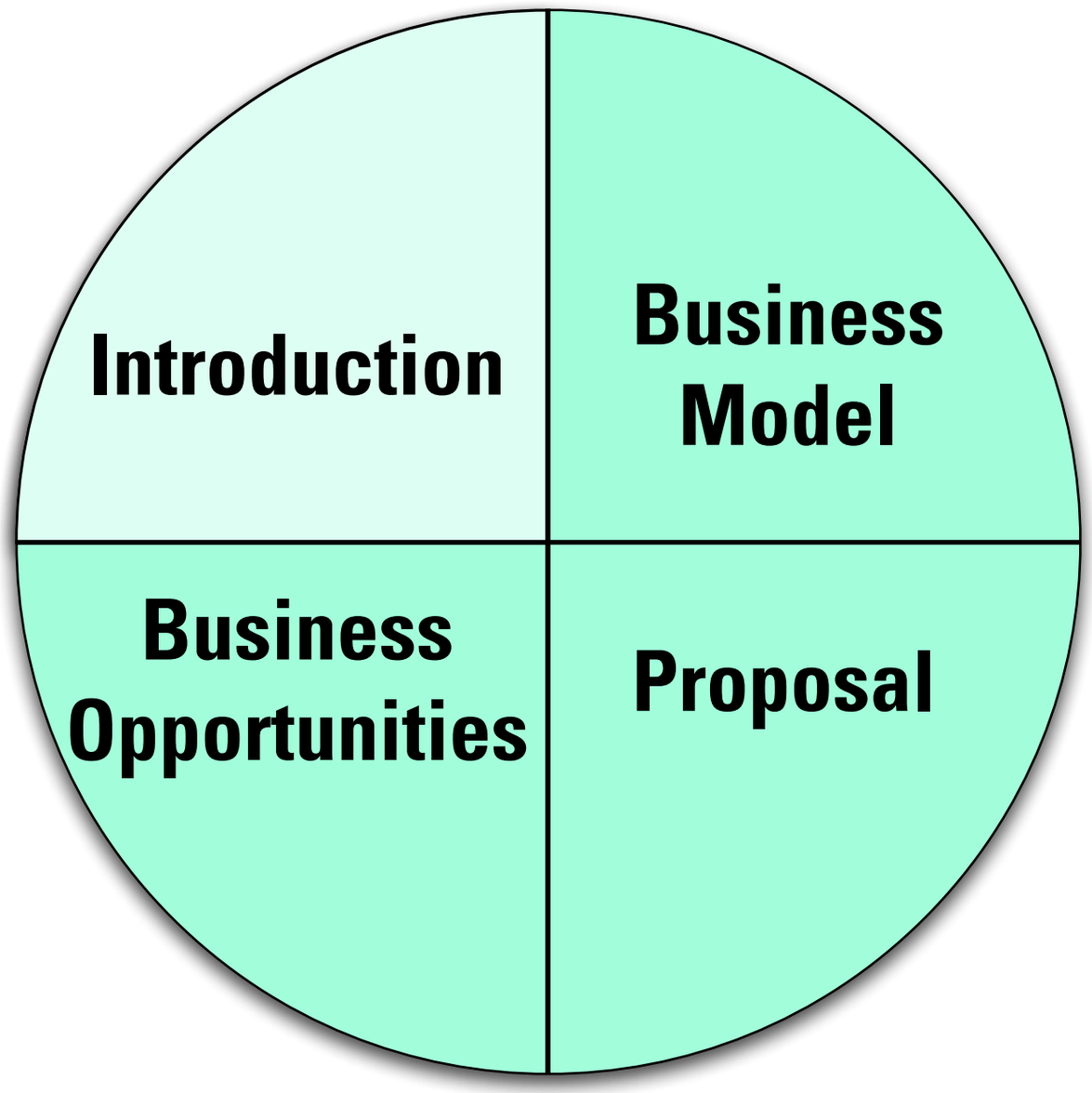
Integrating J2ME Polish into the MTJ Project

ENOUGH SOFTWARE

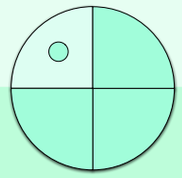




Motivation: Fragmentation

- Mobile device fragmentation limits mobile application adoption and thereby Carriers' ARPU growth
  - Different hardware
  - Different API support
  - Different standards: MIDP, DoJa, BlackBerry, etc.
  - Different interpretations of standards
  - Device bugs

Motivation: Design



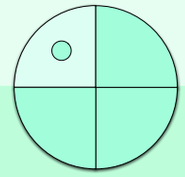
- The mobile Java standard only allows limited design of applications, unless every detail is drawn by the application itself
  - Standard implementations do not allow to influence the CI and look and feel of the application
  - Adapting designs to different environments or customers is difficult
  - Design changes require changing the source code of applications

```
public void paint( Graphics g ) {
   Font font = Font.getDefaultFont();
   g.setFont( font );
   g.setColor( 0x000000 );
   g.drawText( "Hello World", x, y, Graphics.LEFT | Graphics.TOP );
   g.setStroke( Graphics.SOLID );
   g.setColor( 0xFF0000 );
   g.drawLine( x, y + font.getHeight(), x + 200, y + font.getFontHeight() );
}
```

J2ME Polish

- J2ME Polish...
  - overcomes mobile device fragmentation,
  - “polishes” the look and feel of mobile applications
  - eases the development of mobile applications
- J2ME Polish contains a flexible and powerful framework for adapting and designing mobile applications
- Proven Solution:
  - > 120,000 downloads
  - “Pro J2ME Polish” book published by Apress

J2ME Polish



- Selected Features
  - Device Database
  - Create device specific application versions out of a single source code project
  - Resource Assembling: select device and locale specific resources, customize applications easily
  - Localization: static and dynamic localization
  - UI: design applications outside of the source code using simple CSS textfiles - with effects, advanced screens and items, bitmap fonts, etc.
  - Serialization & Persistence framework
  - Java 5 syntax support for J2ME
  - Eclipse integration

Designing Applications

- Design is specified outside of the application's source code
- Customize the application just by modifying CSS
- Example: title { font-color: green; }

Customers

- More than 100 commercial licensees from around the world including:

NOKIA Connecting People

vodafone

MOTOROLA

Sagem Communication SAFRAN Group

COMVERSE

STAR FINANZ

CIRSA interactive

jentro

wayfinder

SPIN

TEMPOS 21 INNOVACIÓN EN APLICACIONES MÓVILES

MetroSpark beta

wavemarket Location Intelligence

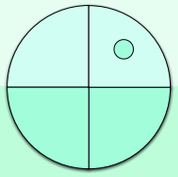
BAMBOO MediaCasting First in Cellular Multicasting

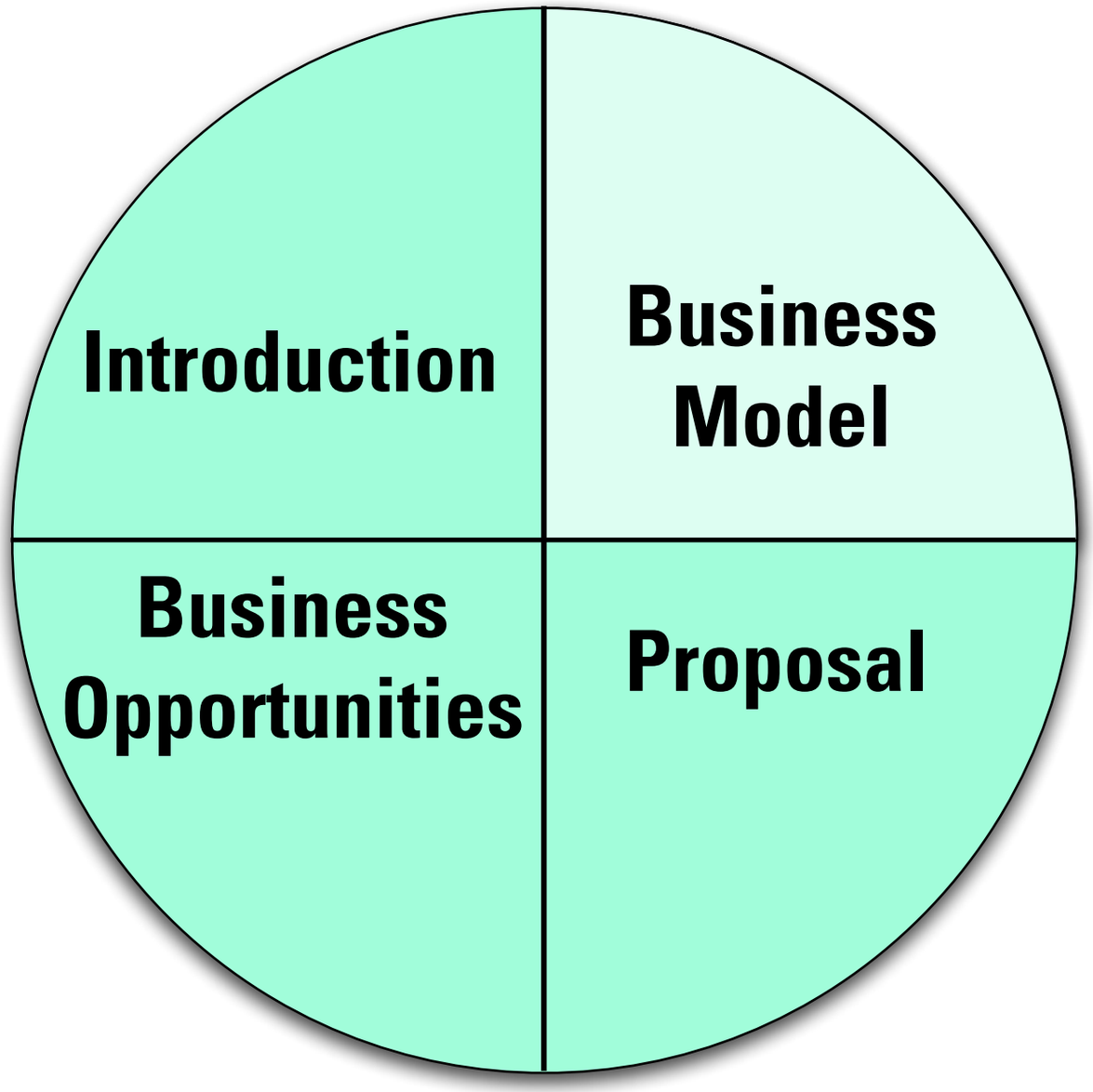
LANCASTER UNIVERSITY

wcities

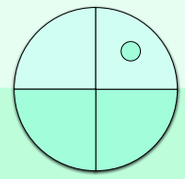
cobramobile : mobile is more™

ENOUGH SOFTWARE



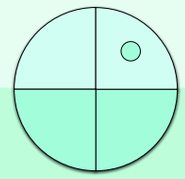


Business Model



- J2ME Polish is dual licensed
  - Open Source GPL license
  - Proprietary commercial licenses
- J2ME Polish can be used commercially under the GPL when only the build framework is used (i.e. preprocessing, resource assembling)
- When J2ME Polish client classes are included into the built applications, the user needs to obtain a commercial license (or publish the application under the GPL)
- Client classes can be used directly (e.g. de.enough.polish.util.HashMap) or indirectly (e.g. when using the J2ME Polish UI framework)

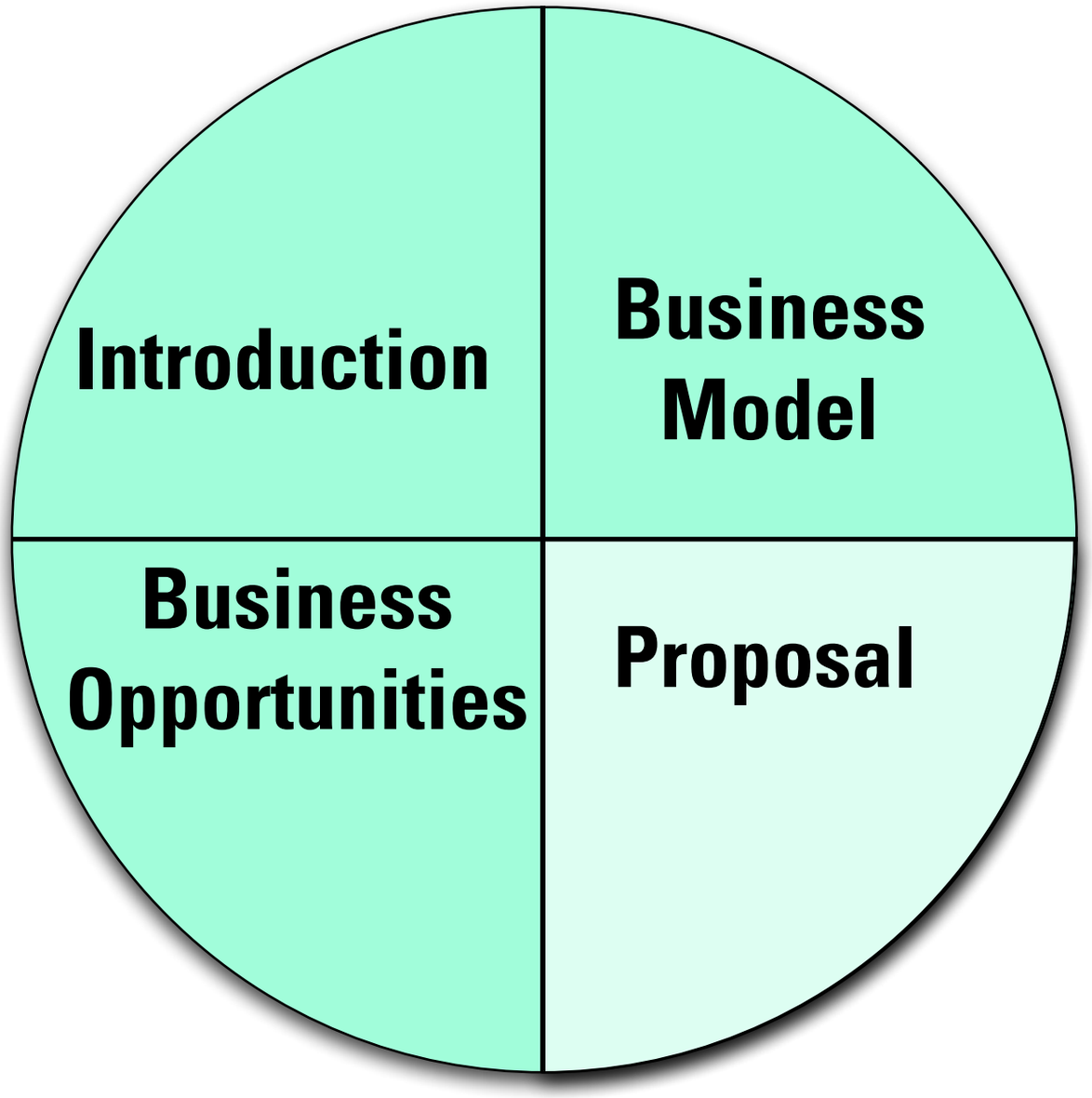
Business Model



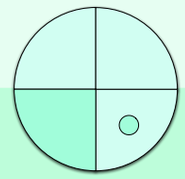
- Providing training and support
- J2ME Polish makes Enough Software known as a specialist for mobile projects and helps to sell services like
  - development
  - porting
  - optimizations
  - GUI improvements

- **Summary**: Interests of Enough Software
  - sell J2ME Polish licenses
  - use J2ME Polish for advertising

ENOUGH SOFTWARE



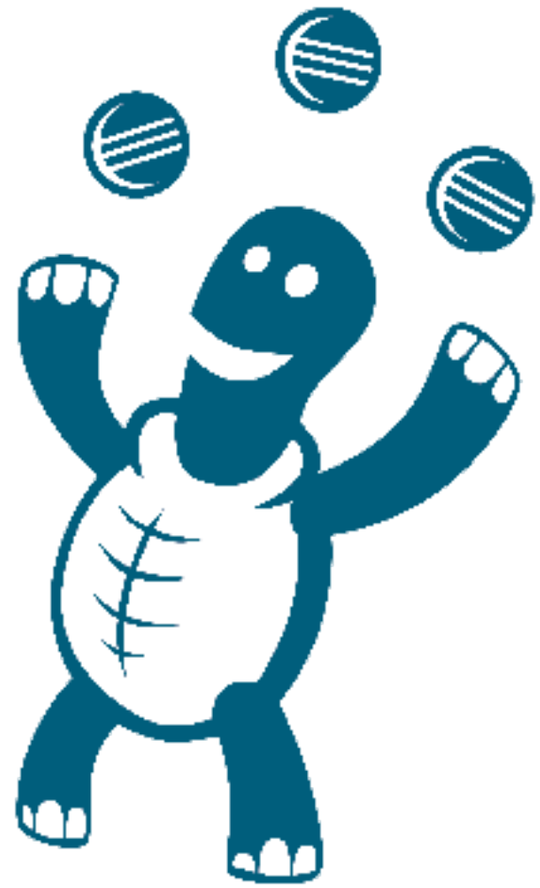
Proposal



- Merge J2ME Polish build features into the MTJ under the EPL license
  - device database
  - device fragmentation
  - localization
  - possibly emulator invocation and debugging
- Cooperation on UI builder
- Working on use cases and implementations
- Providing Resources
  - 1 fulltime resource
  - at least 1 student working on his bachelor thesis for 3 months

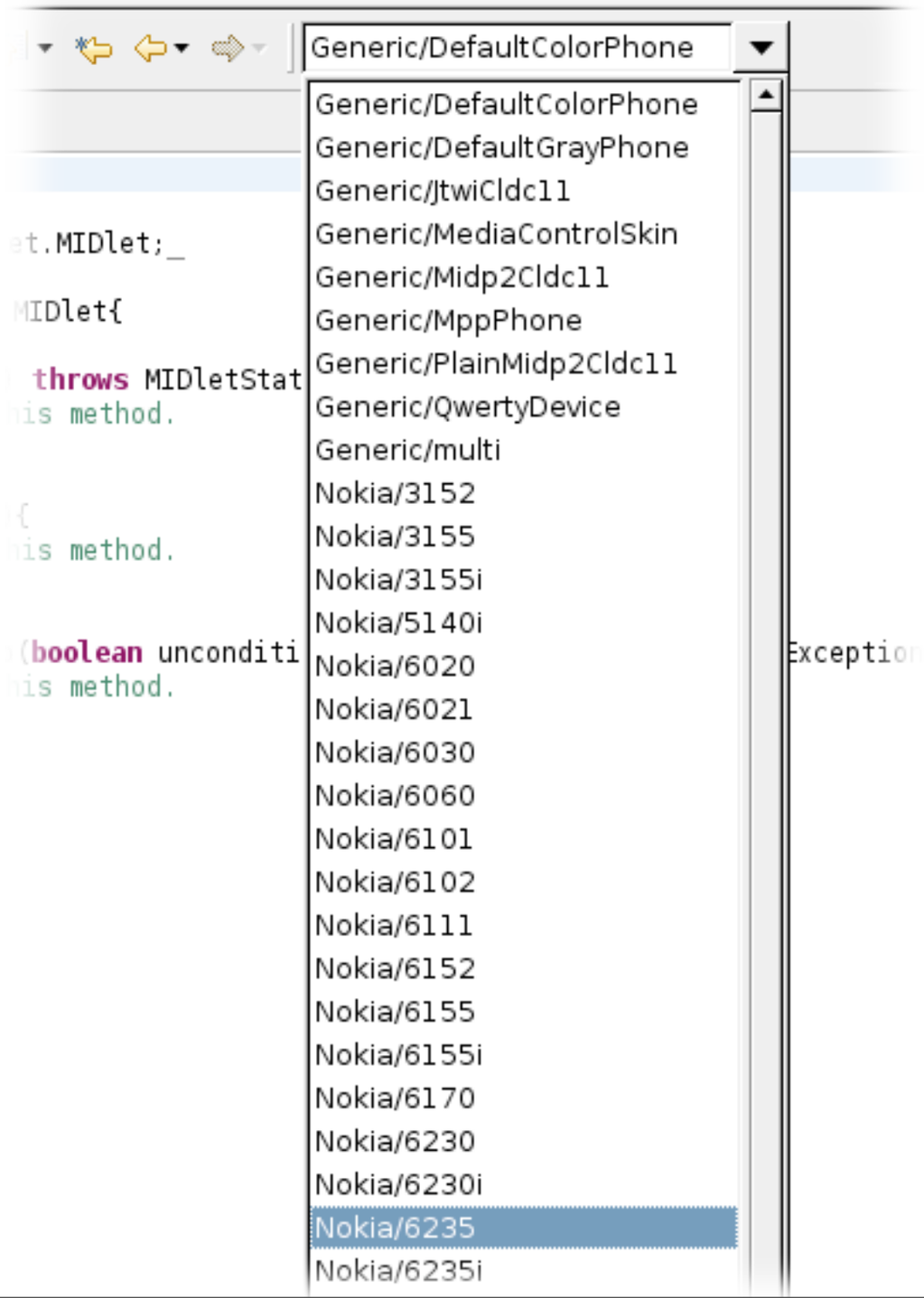
Eclipse Integration

- The J2ME Polish plugin for Eclipse eases the development of mobile applications
  - select target device from drop down menu
  - debugging
  - emulating
  - preprocessing syntax highlighting
  - preprocessing code completion
  - the J2ME Polish build and emulator framework is used for building applications and launching emulators
  - support of any OS: Windows, OS X, Linux
  - plugin can be placed under EPL license



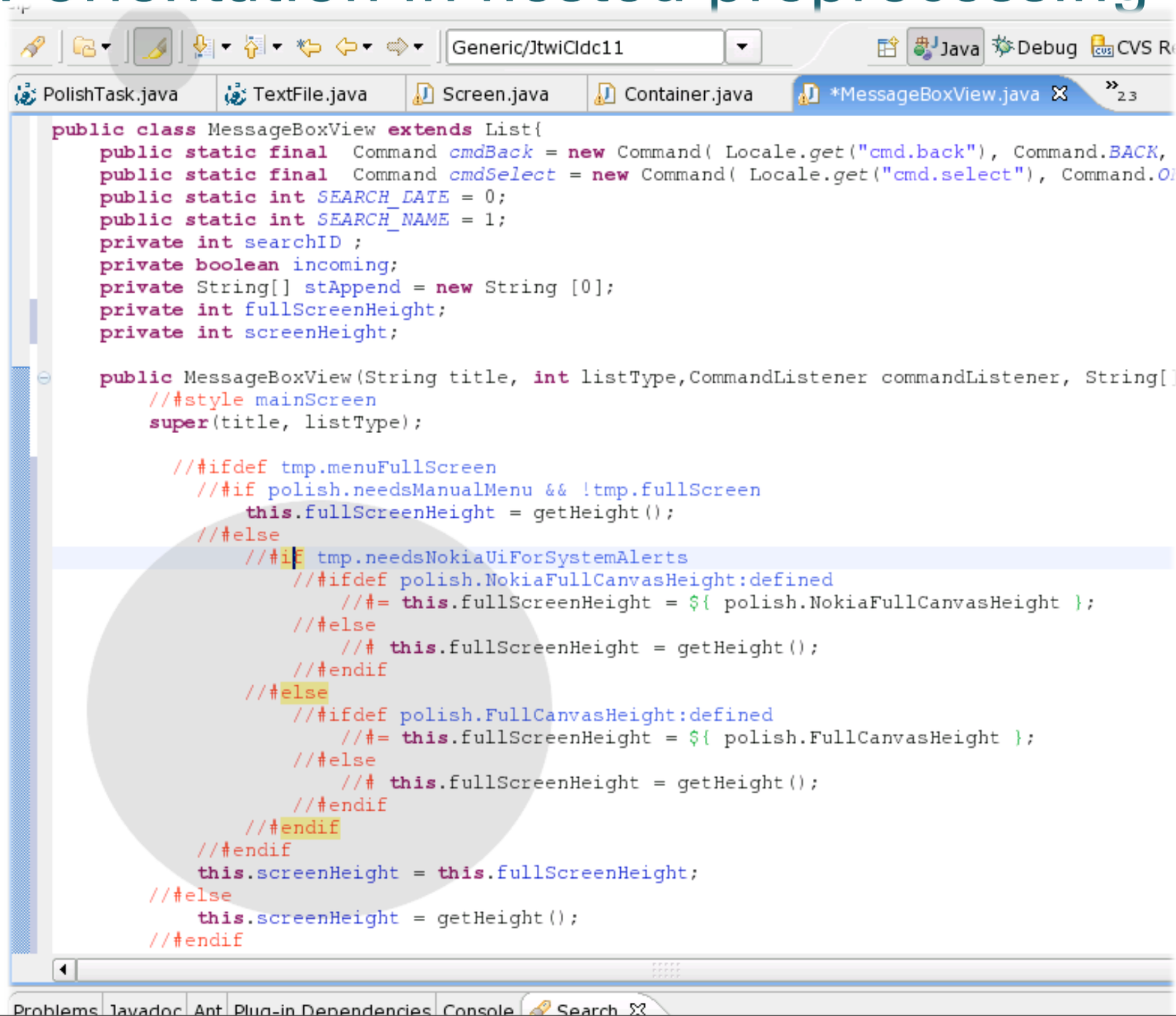
Eclipse Integration

- Target device is selected from drop down menu



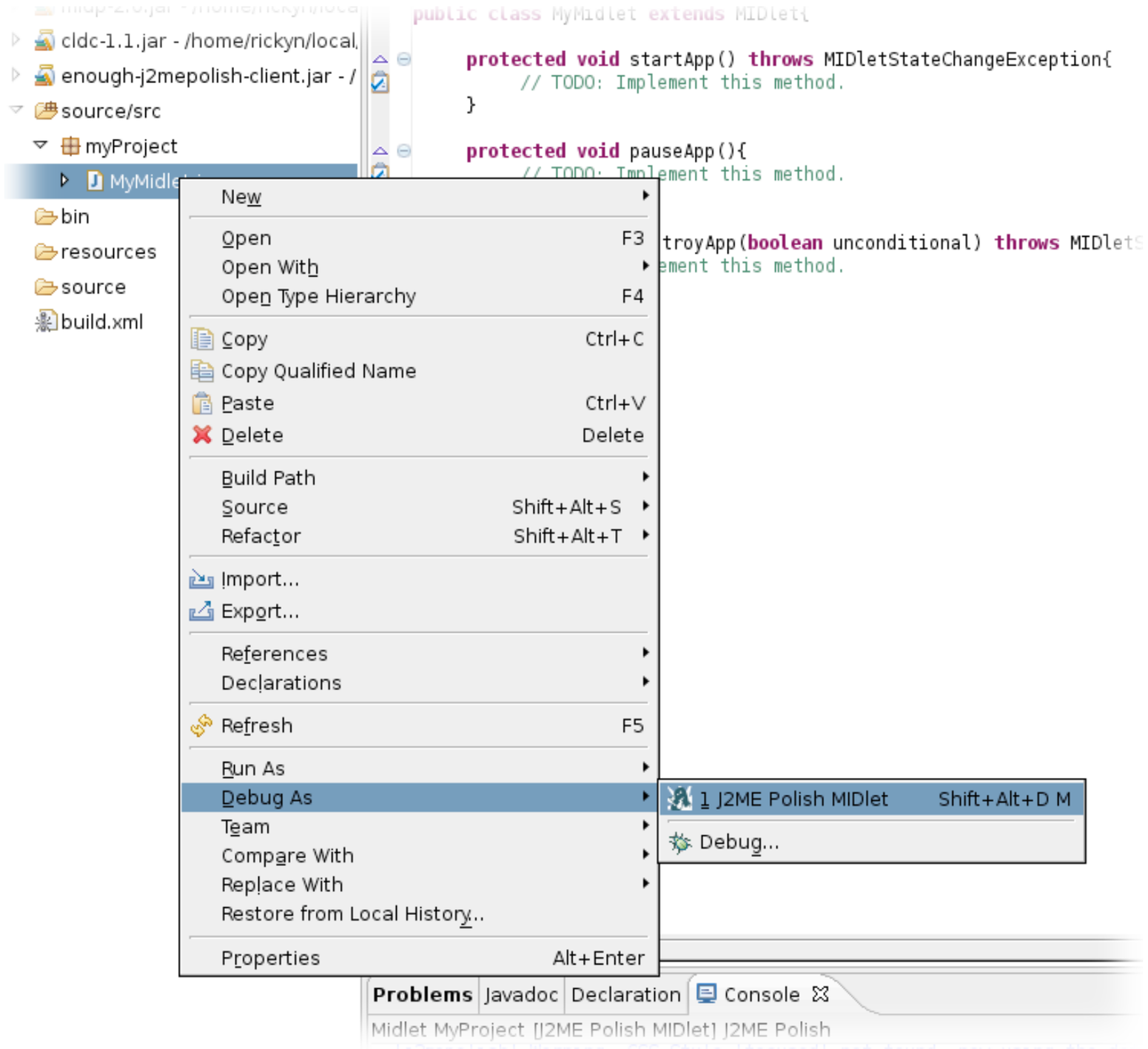
Eclipse Integration

- Syntax highlighting and marking of occurrences allow orientation in nested preprocessing code

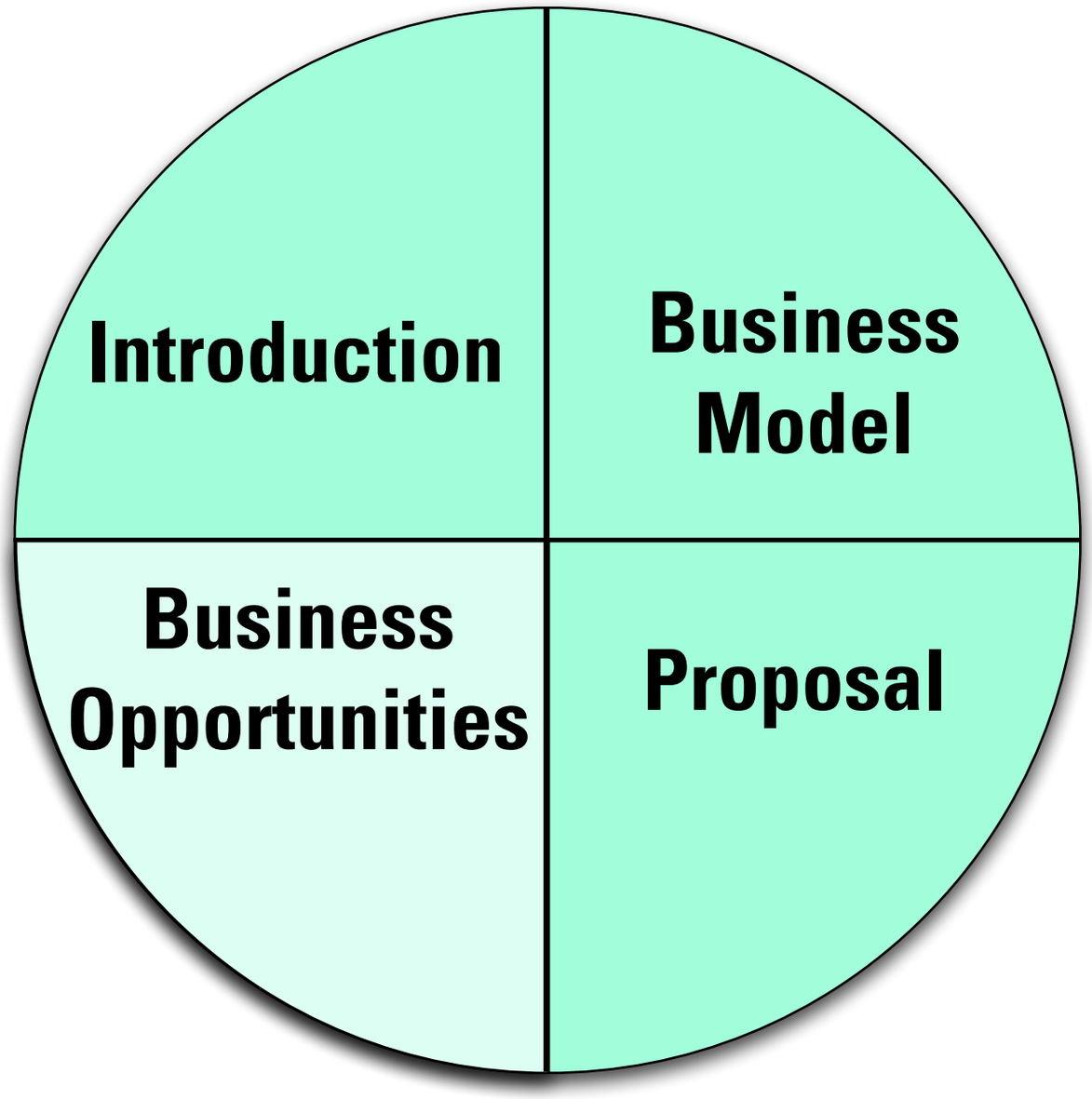


Eclipse Integration

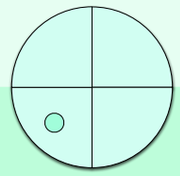
- Debug a project just by launching the debugger



ENOUGH SOFTWARE

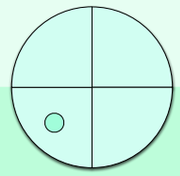


Business Opportunities



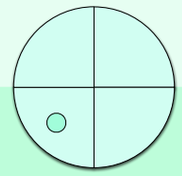
- Main interests of Enough Software are
  - selling licenses
  - selling services by using J2ME Polish as advertising
  - becoming add-on provider for MTJ
- There need to be business opportunities for providing code and resources:
  - Simple integration of J2ME Polish
  - Linking of the J2ME Polish MTJ Product
  - Roadshow MTJ 1.0: speaker/consultant
  - Official MTJ training and integration consultant

Simple integration of J2ME Polish (1/2)



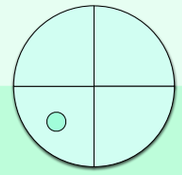
- Idea
  - Some J2ME Polish enhancements like the user interface extensions cannot be released under the EPL without risking our business model; these enhancements cannot be integrated into the official MTJ product for this reason.
  - Users should be able to upgrade the basic MTJ product easily

Simple integration of J2ME Polish (2/2)



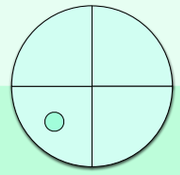
- Proposal
  - Each committer can provide one MTJ extension description containing name, icon, description and update link
  - Upon the first start of the MTJ, the user is asked whether he would like to install one of these upgrades - the upgrade should be possible just by selecting the appropriate checkbox and clicking “upgrade”
  - The user can later onwards install upgrades in the MTJ preferences as well
  - Enough Software helps to integrate this upgrade mechanism

Linking of the J2ME Polish MTJ Product



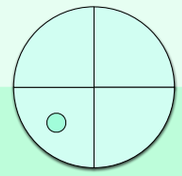
- Idea
  - MTJ based products of committers are linked directly on the MTJ homepage as well as the MTJ download section
  - This would make commercial variants of the MTJ more visable
- Proposal / Background
  - We learned from Jochen Krause, Board of Directors, that there has been a board decision allowing such links as long as they are directly linked to Eclipse products (rather than a company website, for example)

Roadshow MTJ 1.0: speaker/consultant



- Idea
  - Nokia will probably market the MTJ in a roadshow or on Eclipse events, when the 1.0 release is published
  - Enough Software would like to participate on such events as a speaker, thus gaining a business benefit by getting contacts as well as getting paid for the conferences
- Proposal
  - Nokia organizes a roadshow to market the MTJ at least in Europe
  - Enough Software provides a speaker
  - The speaker can inform about commercial extensions such as J2ME Polish

Official MTJ training and integration consultant



- Idea
  - The MTJ homepage should provide a contact list of consultants that help to migrate existing products to the MTJ
- Proposal
  - The MTJ project maintains and publishes a list of consultants that can help with the migration
  - Committers can join this list

Contact

Enough Software
Robert Virkus
Sögestr. 70
28195 Bremen
Germany

Phone +49 - 421 -8409 938
Fax +49 - 421 - 9988 132
Mobile: +49 - 160 - 7788 203

Web http://www.enough.de
Email info@enough.de