Modernize Embedded Linux Software Development Tools to Achieve Development Anywhere

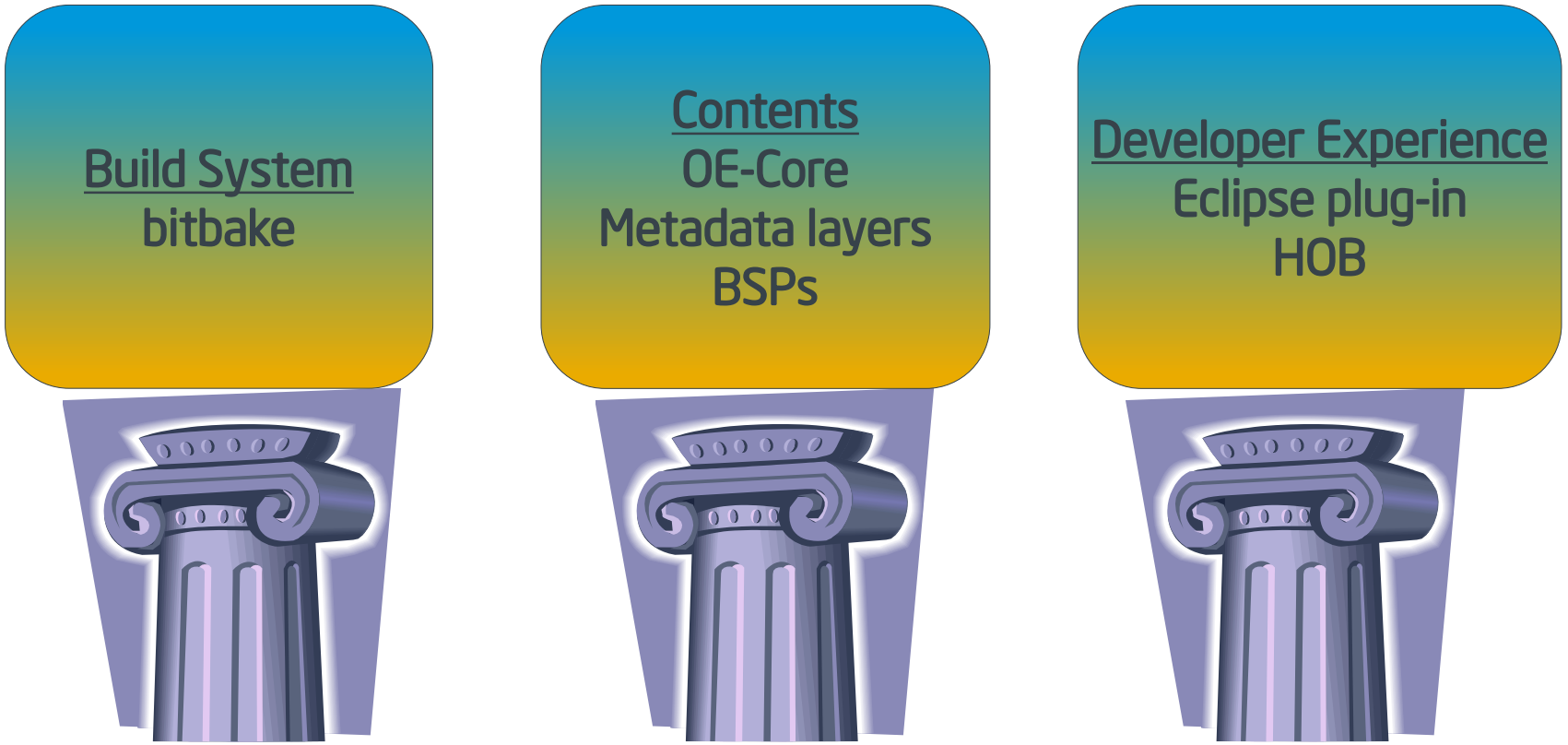
Jessica Zhang

ELCE • Edinburgh • 24 Nov 2013

Agenda

- What are the requirements
- Toaster project
- Eclipse IDE on non-Linux platforms

The Yocto Project Pillars



Continuous Improving

Current Usage model

- **Single user with Linux as development host environment**

Challenges

Better support collaboration among engineers
Able to run and access build out from internet

Support non-Linux based development hosts

Toaster Project

- **Limitation with current Hob**
  - GTK based which is outdated by all means
- **A new web application based project**
  - Fully benefit from latest web technology offerings:
    - modernized look and feel
    - front-end interface via browser achieve platform independent
    - Robust web application frameworks
- **Based on design works from professional design firm and in-house design expert**
  - Feature roadmap, usage flow and user interfaces

Toaster Roadmap and Implementation Approach

- **Freeway First**
  - Needs to be at an advanced stage of maturity to be able to constructively engaging new users to Yocto Project which is unrealistic for the first incarnation
  - Initial iterations will focus on use cases relevant to existing Yocto Project users. This allows to lay the foundation, deliver values to user and attract community contribution early on
- **Inside-Out**
  - Core capabilities – specifying & running builds and conducting image analysis are implemented, refined and stabilized first.
  - Roll out additional features, e.g. project support for collaboration, public Toaster, etc.

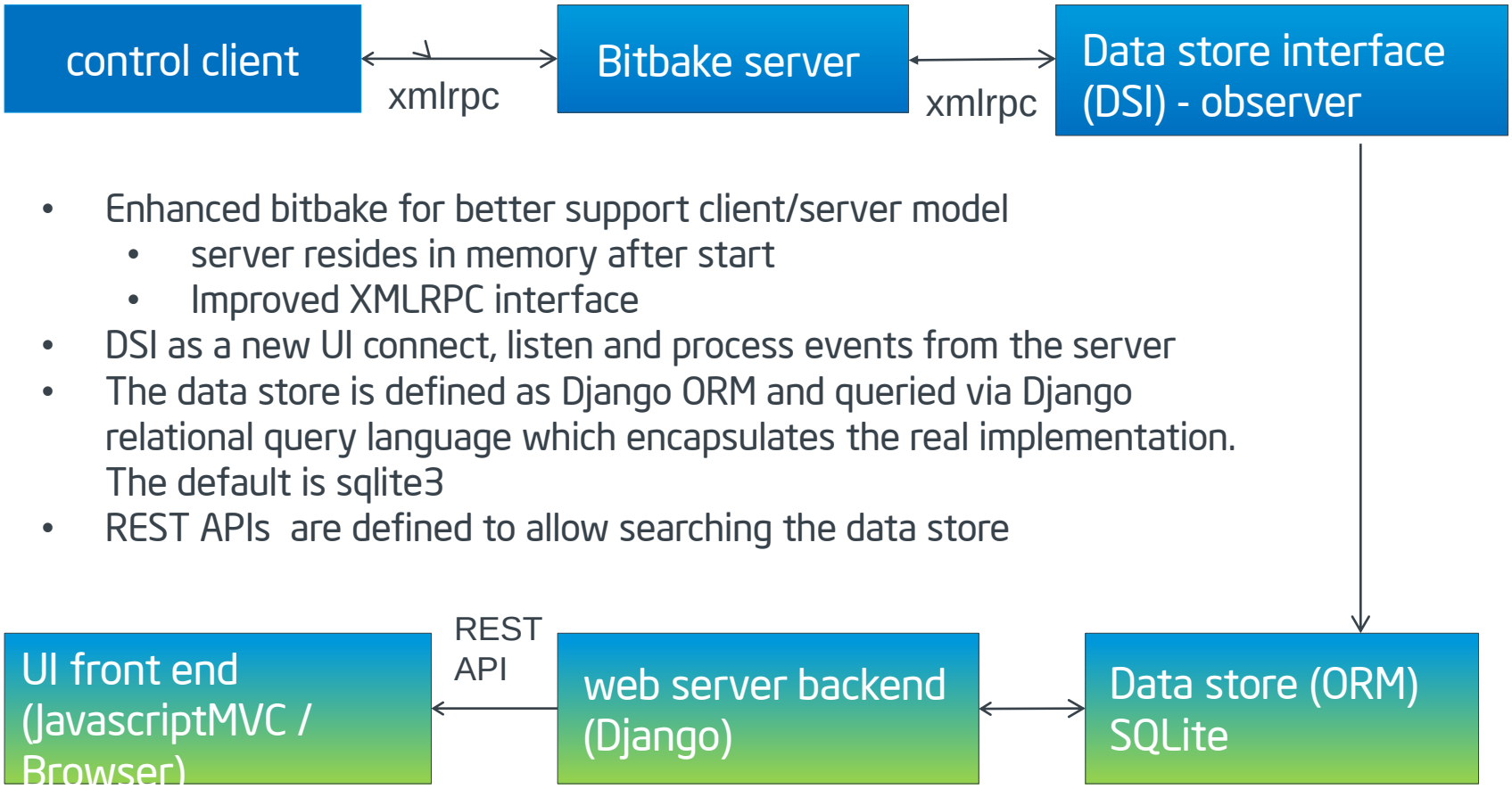
Toaster Roadmap Overview

Overview Diagram

tobias & tobias

		STAGES							
		ONE	TWO	THREE	FOUR	FIVE	SIX	SEVEN	EIGHT
S10	Create Public Web Hob Account				TARGET				
S11	Manage PWH Account				MINIMAL		TARGET		
S05	Create Build		MINIMAL	TARGET	REFINE	REFINE	REFINE	REFINE	REFINE
S25	Export Project				MINIMAL	TARGET	REFINE		REFINE
S30	Show Build Metrics	MINIMAL		TARGET		REFINE			
S31	Error Messaging			MINIMAL	TARGET		REFINE		REFINE
S35	Finding/Viewing Builds	MINIMAL			TARGET			REFINE	
S40	Install Single-User Web Hob	MINIMAL					TARGET		
S45	Install Web Hob Server		MINIMAL			TARGET			
S55	Administering Accounts					MINIMAL		TARGET	
S70	Import Project				MINIMAL		TARGET		REFINE
	DIRECTION	EXTEND	EXTEND	ENHANCE	EXTEND	ENHANCE	EXTEND	ENHANCE	ENHANCE
	OBJECTIVES	Local Web Hob Exists; Initial Release of Web Hob	Team Web Hob Exists	Achieve Target state for core capabilities	Public Web Hob Exists	Support scaling of Team Web Hob	Easier migration for non-experts	Completion of planned roadmap	

Toaster General Design



- Enhanced bitbake for better support client/server model
  - server resides in memory after start
  - Improved XMLRPC interface
- DSI as a new UI connect, listen and process events from the server
- The data store is defined as Django ORM and queried via Django relational query language which encapsulates the real implementation. The default is sqlite3
- REST APIs are defined to allow searching the data store

Early Access To Toaster

- **Wiki page: https://wiki.yoctoproject.org/wiki/Toaster**
- **Code:** http://git.yoctoproject.org/cgit/cgit.cgi/poky-contrib/?h=webhob-poky/master

Running Toaster

- **Setup build environement:**

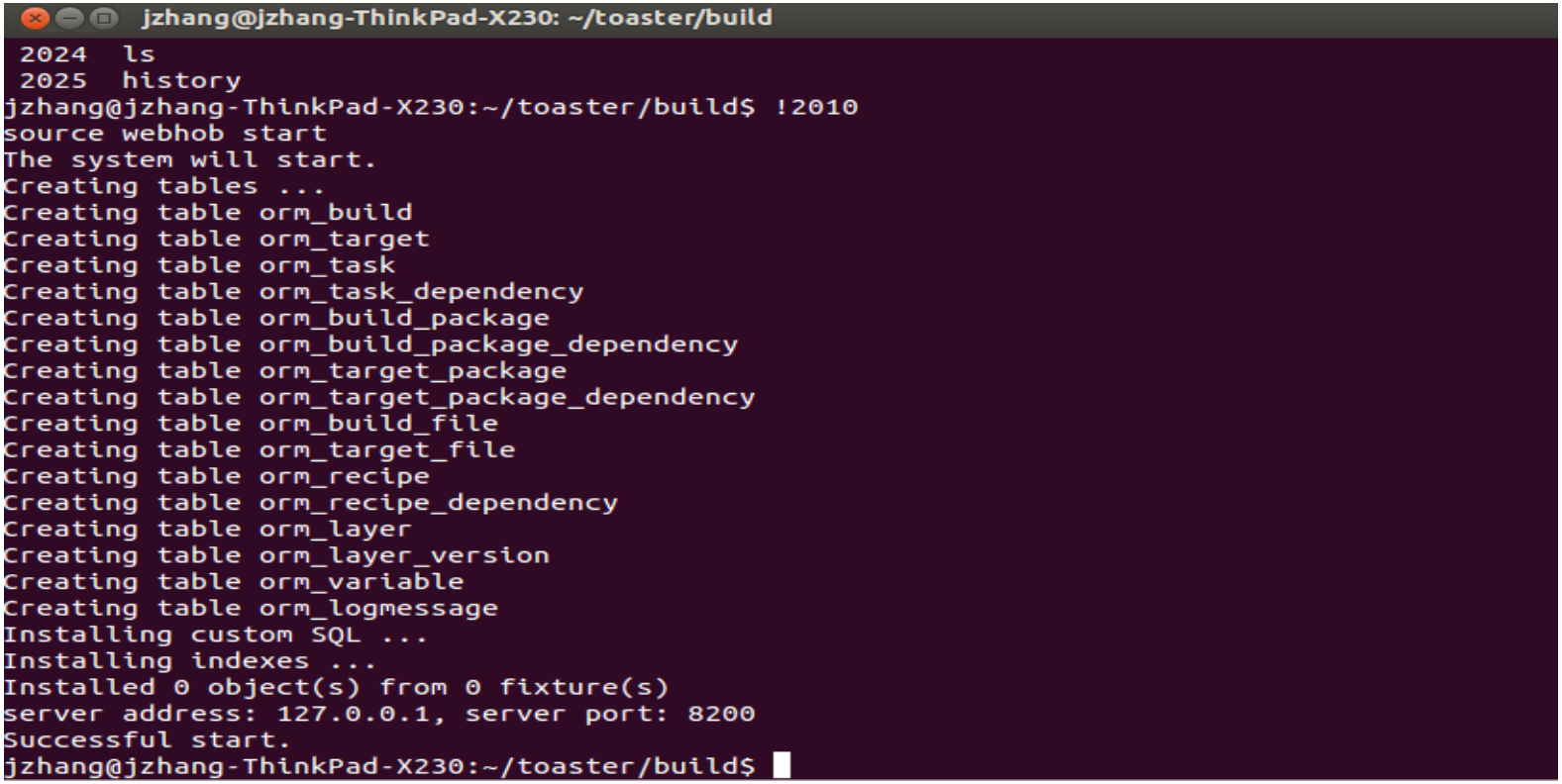
```
$ source oe-init-build-env
```

- **Build history enabled in conf/local.conf**
  - INHERIT += “buildhistory”
  - BUILDHISTORY_COMMIT = “1”

Running Toaster (2)

- **Start toaster server**

  `$ Source toaster start`



Running Toaster (3)

- **Running a build**

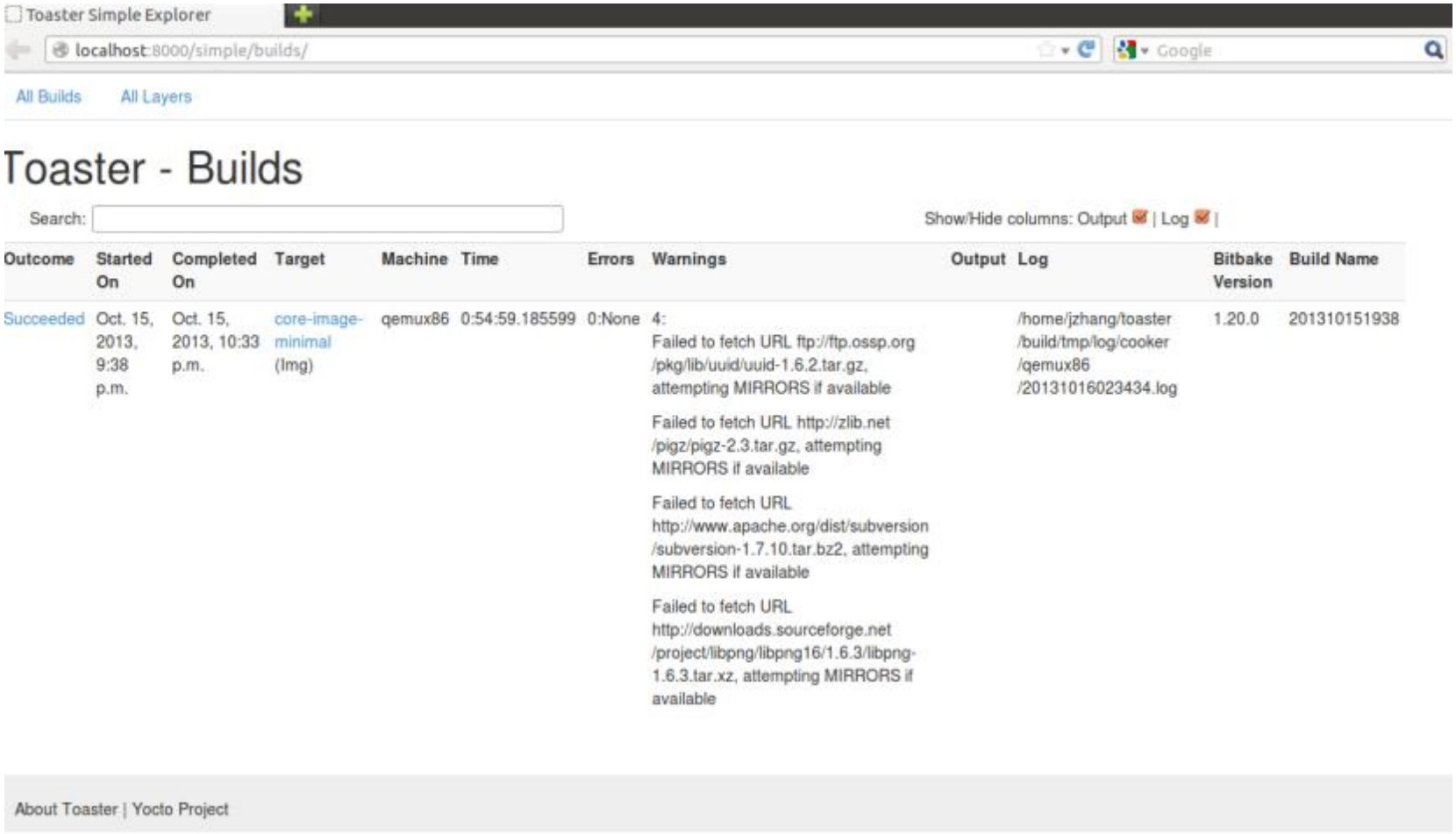
  `$ Bitbake core-image-minimal`

```
jzhang@jzhang-ThinkPad-X230: ~/toaster/build
Successful start.
jzhang@jzhang-ThinkPad-X230:~/toaster/build$ bitbake core-image-minimal
Parsing recipes: 100% |##########################################################| Time: 00:00:44
Parsing of 855 .bb files complete (0 cached, 855 parsed). 1186 targets, 35 skipped, 0 maske
d, 0 errors.
NOTE: Resolving any missing task queue dependencies

Build Configuration:
BB_VERSION           = "1.20.0"
BUILD_SYS            = "x86_64-linux"
NATIVELSBSTRING      = "Ubuntu-12.04"
TARGET_SYS           = "i586-poky-linux"
MACHINE              = "qemux86"
DISTRO               = "poky"
DISTRO_VERSION       = "1.5"
TUNE_FEATURES        = "m32 i586"
TARGET_FPU           = ""
meta
meta-yocto
meta-yocto-bsp       = "(nobranch):f5896b40ced9a7ebbd7428f8792dd063cd095bfb"

NOTE: Preparing runqueue
NOTE: Executing SetScene Tasks
NOTE: Executing RunQueue Tasks
WARNING: Failed to fetch URL ftp://ftp.ossp.org/pkg/lib/uuid/uuid-1.6.2.tar.gz, attempting
MIRRORS if available
WARNING: Failed to fetch URL http://zlib.net/pigz/pigz-2.3.tar.gz, attempting MIRRORS if av
ailable
WARNING: Failed to fetch URL http://www.apache.org/dist/subversion/subversion-1.7.10.tar.bz
2, attempting MIRRORS if available
WARNING: Failed to fetch URL http://downloads.sourceforge.net/project/libpng/libpng16/1.6.3
/libpng-1.6.3.tar.xz, attempting MIRRORS if available
NOTE: validating kernel config, see log.do_kernel_configcheck for details
NOTE: Tasks Summary: Attempted 1584 tasks of which 255 didn't need to be rerun and all succ
eeded.
```

Example Simple Interface



What's Next

- **Initial release for early trial is targeted for 1st week of Nov.**
  - Supported bitbake changes
  - Toaster backend framework
  - REST APIs
- **Define feature deliveries for 1.6 based on the usage scenarios with the "inside-out" approach with frontend support**

Eclipse IDE on non-Linux Platforms

- **Remove the barrier that embedded Linux development must be on a Linux platforms:**
  - Corporate IT setup
  - Developer personal preference
- **Fundamental building blocks**
  - Java network URI class encapsulates key URI information:
    - host, port, path, scheme, etc.
  - Eclipse Target Management (TM) project's Remote System Explore(RSE) framework various pluggable remote resource subsystems:
    - Remote files subsystem, Shell, Process, Terminal, etc.

Eclipse Community Prototypes

- **Eclipse CDT (C/C++ Development Tooling)**
  - Has the capability to create a project on remote location
  - Can’t do configuration for autotool based project
  - Can’t build
- **Eclipse Parallel Tools Project (PTP) Remote Development Tools (RDT) feature**
  - Aims at provides a framework and reference implementation that facilitates using a local IDE to perform development tasks on another, remote machine
  - Cumbersome to setup, not stable, no autotools support, not follow CDT model

Yocto Project Eclipse Plug-in Changes

- **Code refactor and created common remote.util plugin**
  - facilitates remote operations using RSE remote file, shell, process and terminal subsystems
- **Made contributions to upstream**
  - Fixed deadlock situation
  - Allows asynchronous operation which makes remote operation more robust
- **Replace IPath based location specification with URI**
  - IPath only allows local file system access

Yocto Project Eclipse Plug-in Changes (cont)

- **Add remote location specification to project template and preference setup**
- **Able to create bitbake commander project and ADT project that extend cdt autotool based project remotely**
  - Modified the code by changing some of IPath usage to be LocationURI based.
  - The remote projects still need to be created on Linux machines

  1 build can only run on Linux platforms

  2 cross toolchain and sysroot also requires Linux environment

Remaining Challenges – System Development

- **To run Yocto Build against bitbake commander project**
  - Either via build appliance virtual machine that runs on the same non-Linux platform
  - Or still need to access the remote Linux machine
- **Till Toaster is ready that allows to run the build via web browser**

Remaining Challenges – Application Development

- **Remote Yocto Project autotool projects cross compilation needs to happen on the remote Linux host.**
- **The core implementation of CDT build call stack, e.g. CommonBuilder, BuildRunHelper and ManagedBuidManager are tightly coupled with the usage of IPath.**

Remaining Challenges – Application Development (2)

- **For switching to use URI based location,**

  Option 1: change the overall CDT code base

  It is not supported by the CDT upstream

  Option 2: fork the CDT implementation

  - Only make needed changes to support our usage model,
  - Route the call via the customized builder registration/extension.
  - Long term maintainance and sync issues with upstream CDT

Modernize Embedded Linux Software Development Tools to Achieve Development Anywhere

Modernize Embedded Linux Software Development Tools to Achieve Development Anywhere

Modernize Embedded Linux Software Development Tools to Achieve Development Anywhere

Thank you for your participation!