

OS Tools for OpenBSD

Overview Presentation

Team Fugu

CS 486 - Capstone

Team Fugu

- Ben Atkin
- Thad Boyd
- Nauman Qureshi
- Erik Wilson

Fugu: A poisonous blowfish.
The blowfish is the OpenBSD mascot.



Sponsor

- US Geological Survey (USGS)
 - Ernest Bowman-Cisneros and Margaret Johnson.
 - System Administrators for Astrogeology Team.

Task Definition: Problem

➤ Our Client

- USGS Astrogeology Division.
- Data Analysis, Image Processing, etc.
- Heavy Demands On Computing Infrastructure.
- Heterogenous Computing Environment.

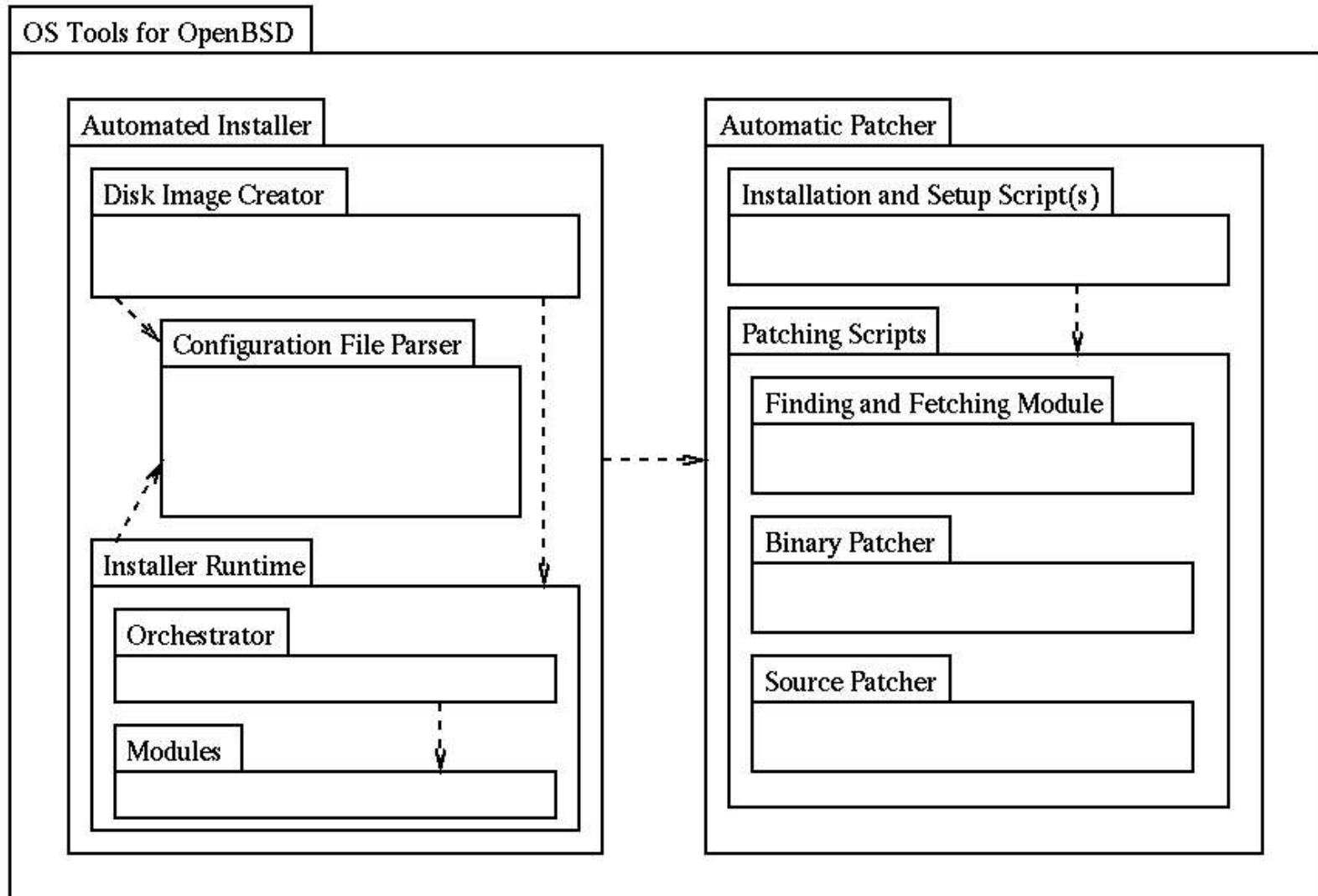
➤ The Problem

- Client uses Automatic Installers and Patchers to save time and improve consistency, security.
- Client uses OpenBSD, a free and secure OS.
- OpenBSD lacks suitable Automatic Installer and Patcher tools.

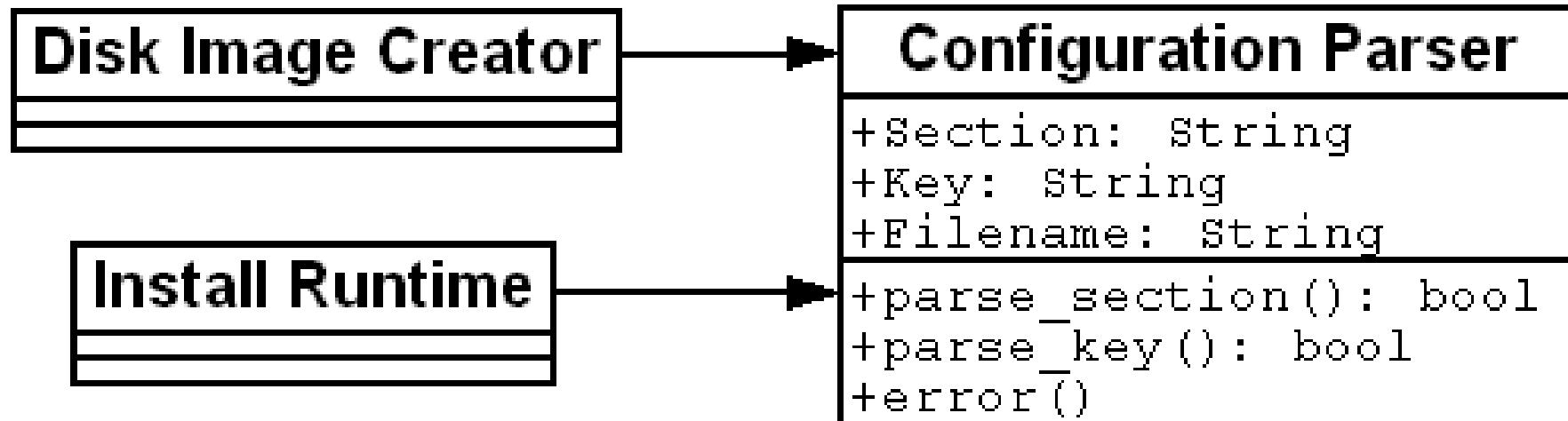
Design Methodology

- Using pre-existing code as much as possible.
- Modularized Tepatche into Source and Binary Modules.
- Add various utilities such as config parser and logging facilities.
- Created well defined architecture.

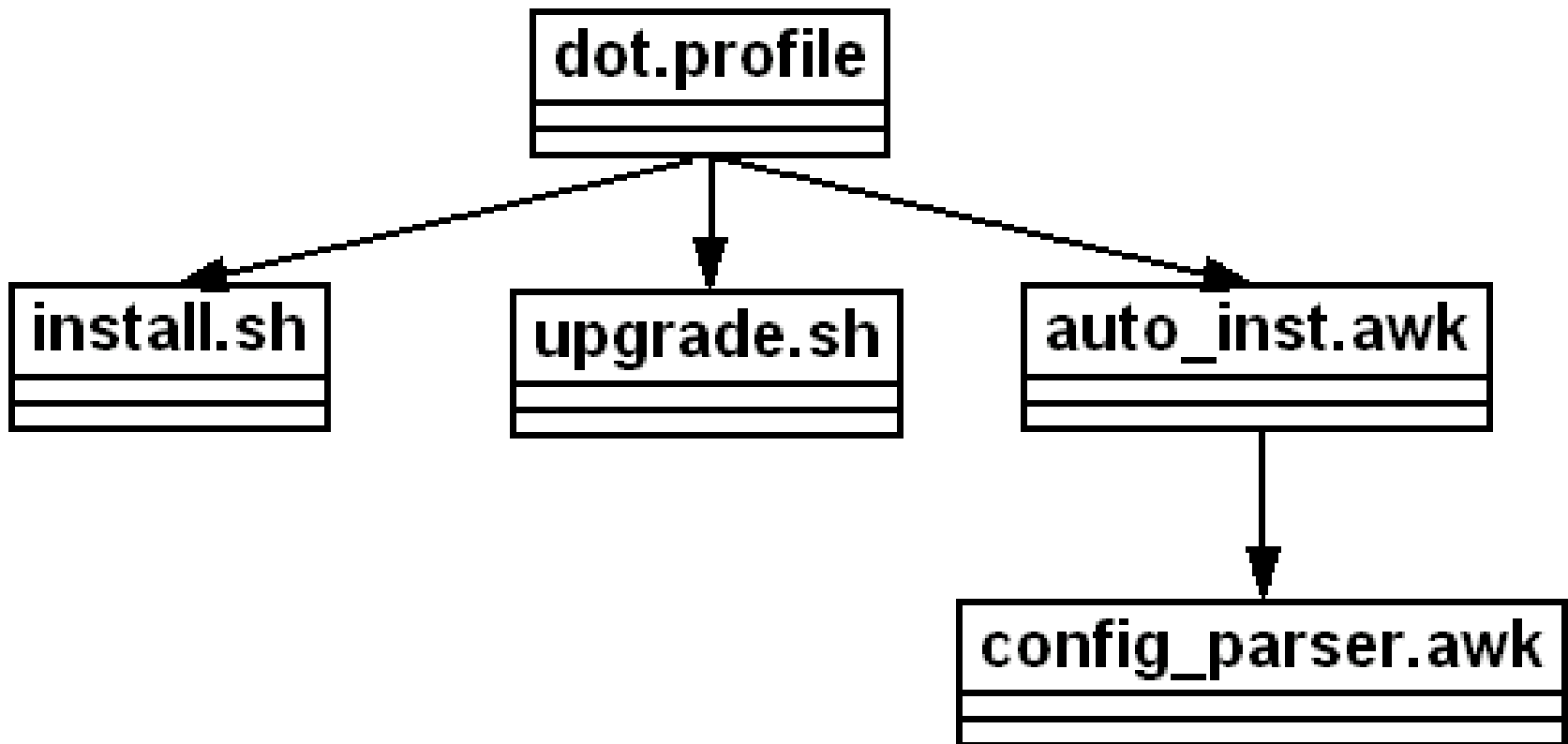
Architecture: Overview



Architecture: Install Interactions



Architecture: Install



Architecture: Shell Scripting

- Use global environment variables
- Advantage: can be accessed inside backquotes
- Backquotes: execute a command string, and capture the output
- Very few tools at our disposal, because the installation needs to run from a floppy
- Disk space calculations must be done in a shell script.

Architecture – Disk Setup

➤ Disk Partitioning

- Requirement: Must be able to specify partition sizes proportionally
- Solution:
 - Required Size – has priority
 - Desired Size – extra space divided proportionally
 - Extra Space – beyond the desired size, the extra space can be assigned to a partition

➤ Two kinds of partitions

- FDISK (on i386 only)
- disklabel

Architecture – Package Installation

➤ Installing Packages

- OpenBSD Filesets
- Extract to / once the packages are on disk
- If they're on the network, copy to /tmp
- Copy, install, delete, before getting next package, so /tmp won't run out of space

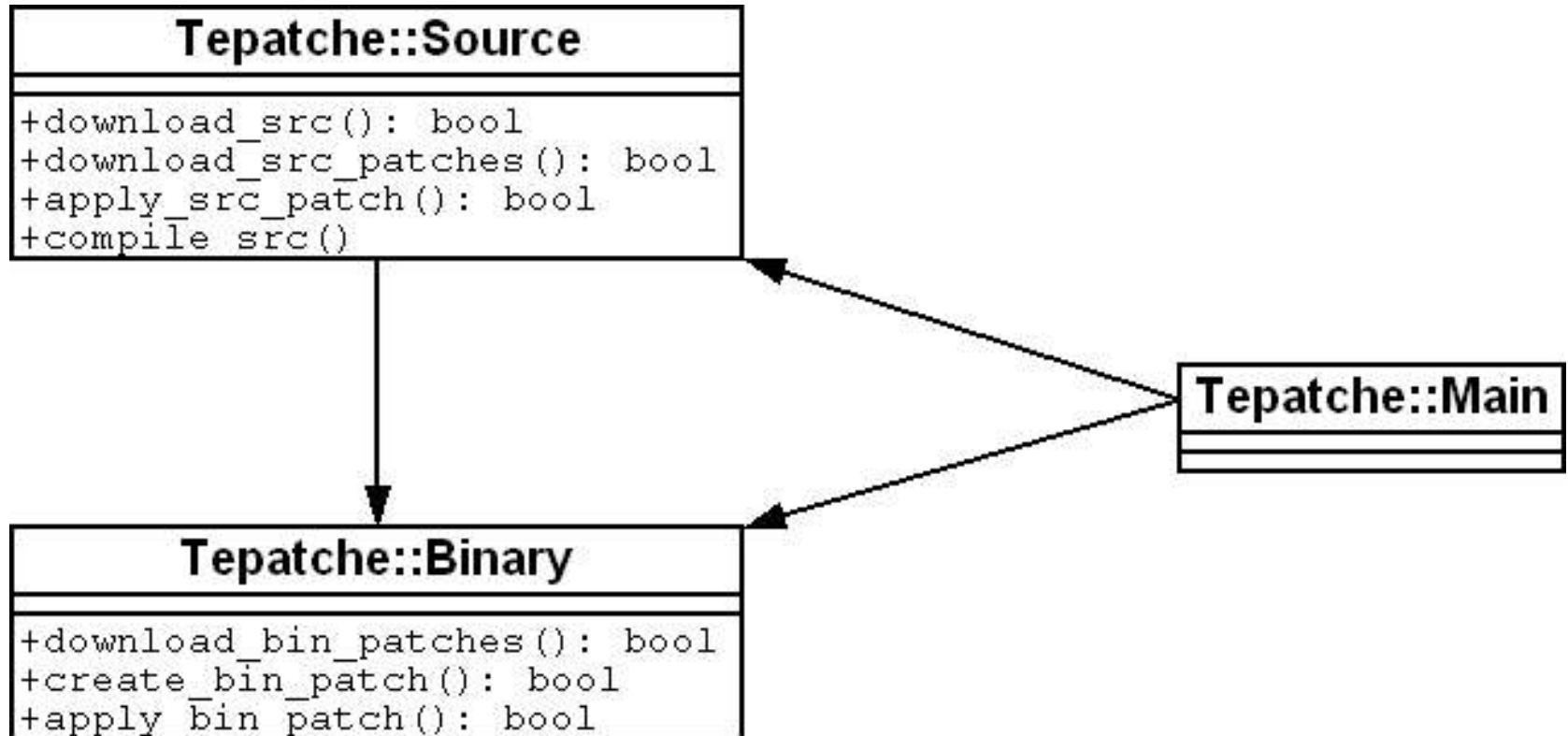
Architecture: Completing Install

- Permanently write settings to disk
 - Disk mounting: fstab/mstab
 - Networking
 - Time/Date
- Post-Install
 - Run script in configuration file
- No interesting screenshots yet
- Significant portions are complete

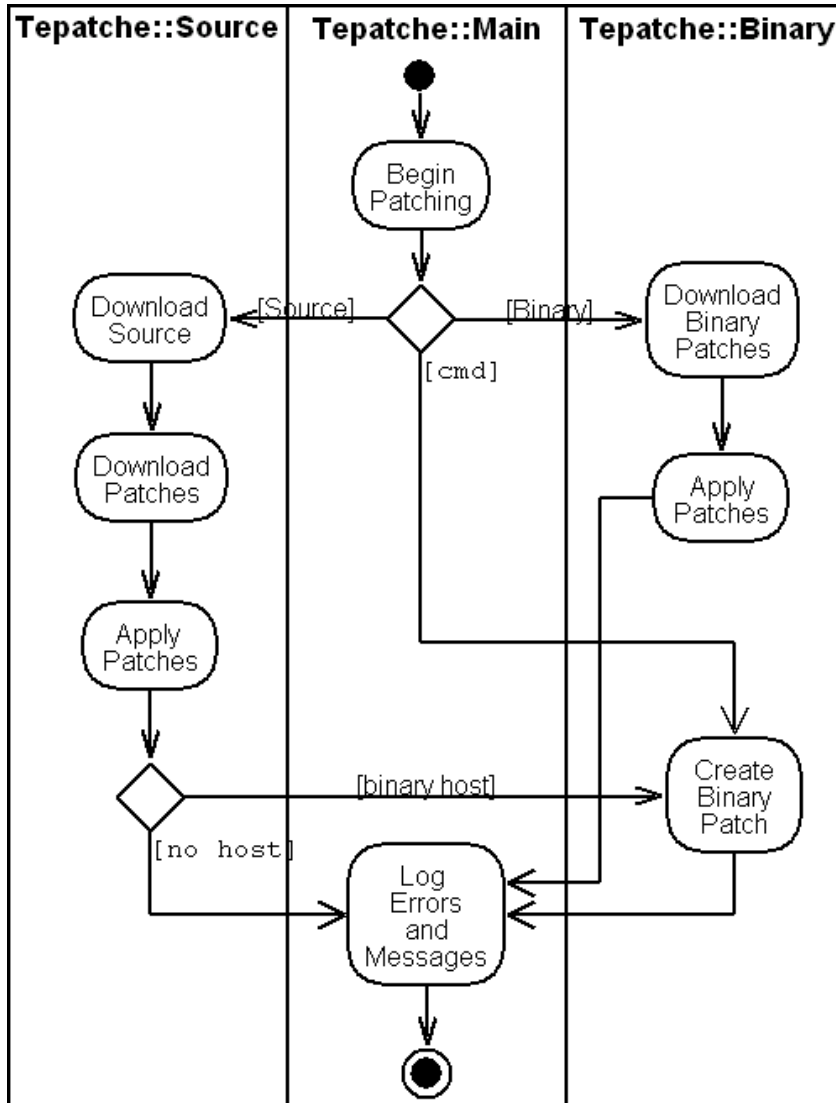
Automated Patcher

- Patcher will be created by adding new features to the existing Tpatche Perl Script.
- The script will be divided into three main modules for better organization of functions.

Design Overview



Automated Patcher



Functionality:

- Download Source
Tarballs if needed
- Download Source and
Binary Patches as needed
- Application to the system of
those patches
- Create binary patch given
that the source is patched

Main Tepoche Module

- Process:

The main module will provide the interaction from the shell and orchestrates all of the resulting actions

Main Tepoche Module

- Functionality:

- Read information from the “config” file
- Call the appropriate module (Source or Binary) to perform patching
- Send the appropriate messages and errors to location defined in the config file
- Schedule a reboot if specified in the config file

Tepatche Source Patching Module

- Process:

The source module will be used to compile from source any patches which exist from the OpenBSD website

Additionally it may act as a server by creating Binary Patches from the compiled source

Tepatche Source Patching Module

- Functionality:

- Download the source tarballs and deflate if source does not exist on the system
- Download patches from the OpenBSD web server, or the other defined location
- Apply the downloaded patches to the source
- Compile the source
- Creating a Binary Patch if desired from the compiled source patches

Binary Patching Module

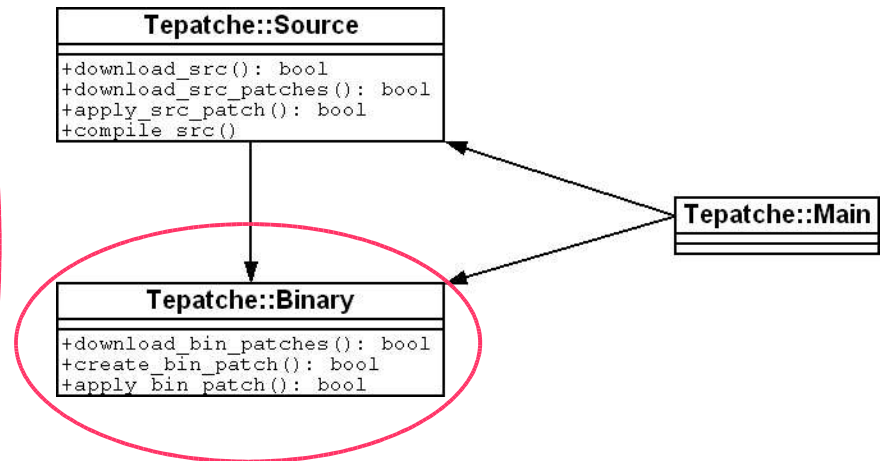
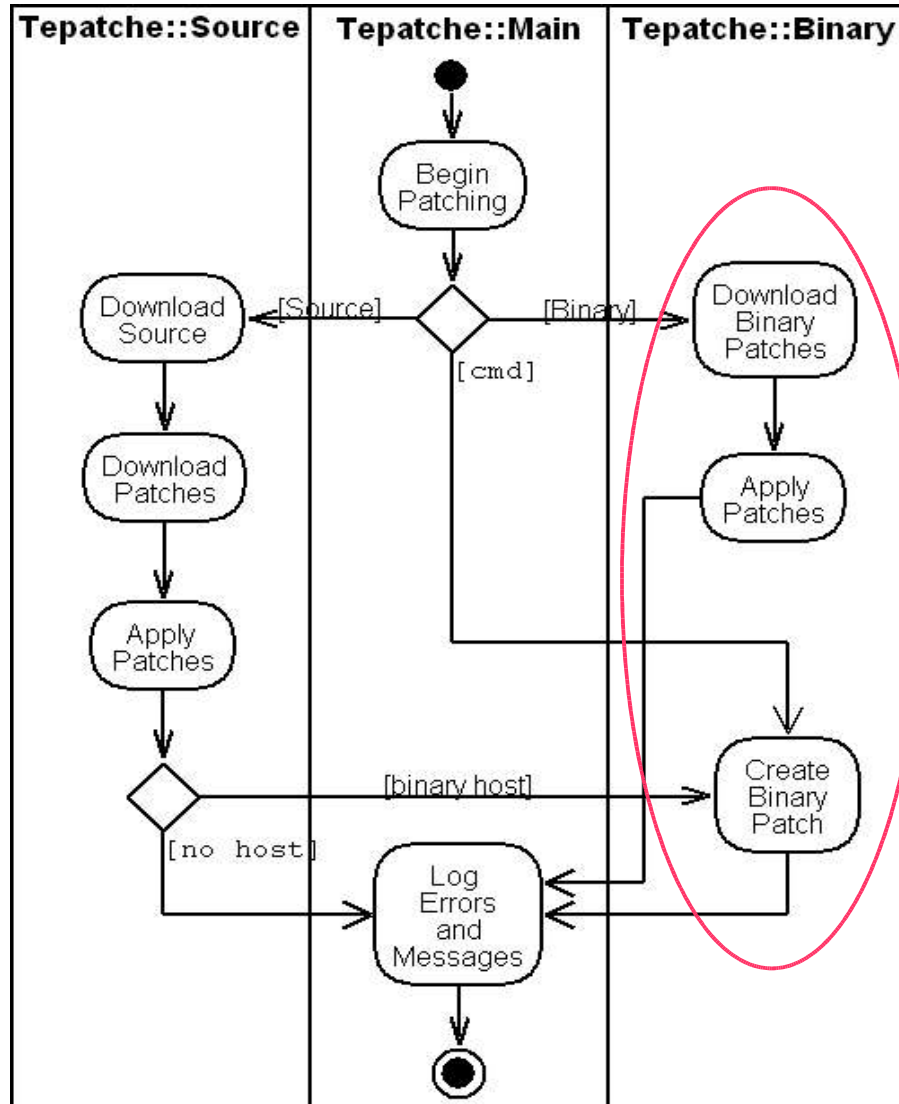
- Process:

This module will be useful for avoiding the slow compiling speeds associated with source patching

Binary Patching Module

- `create_bin_patch`
 - create `.tgz` package using `pkg_create`
- `download_bin_patches`
 - download patches from location specified in config file
- `apply_bin_patch`
 - install `.tgz` package using `pkg_add`

Tepatche Binary Packaging: Architecture



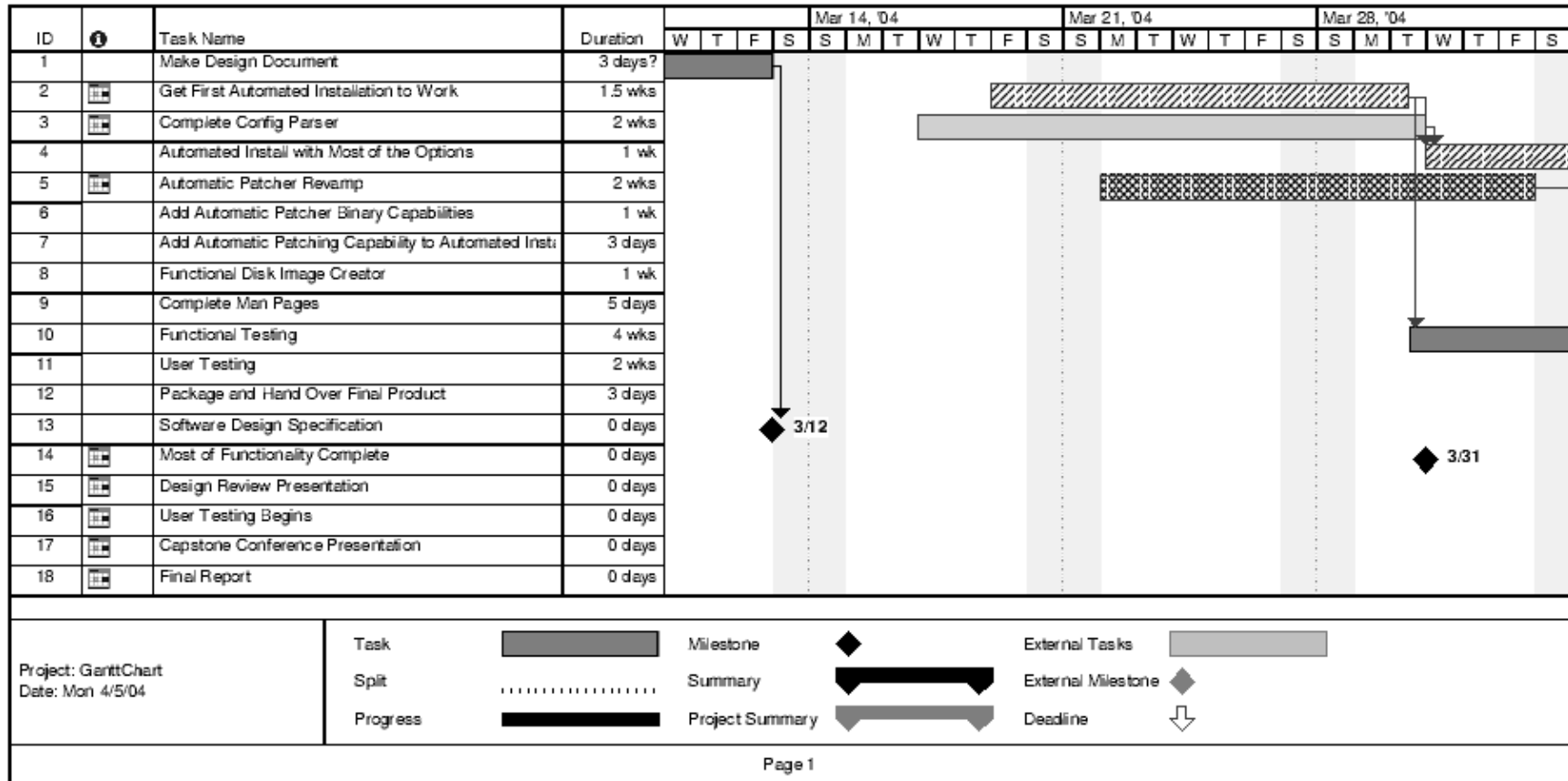
Binary Patching Module

- Functionality:
 - Download Binary modules of a specified architecture hosted on the network
 - Apply those binary patches using the *pkg* facility

Features to Add

- Log all package installation history
- Roll back existing package; do not attempt to download and install again
- Choose which packages to install and which not to
- Install package from non-FTP location
 - HTTP
 - NFS

Tasks: Completed



Major Tasks and Deadlines

- 2004.04.09 (Friday)
 - Combine/Swap Projects
- 2004.04.16 (Friday)
 - Complete installer testing
 - Complete Tepatche testing
- 2004.04.19 (Monday)
 - Complete final presentation
 - Give project to USGS for testing and feedback

Conclusions

- Automated Install almost completed.
- Focused on learning PERL.
- Lots of attention on team work.
- Demonstration.