

Pluggable type-checking

- An application of Java SE 8 type annotations
- Java's type-checker prevents errors
 - `int x = "hello world";`
- Java's type-checker doesn't prevent **enough** errors
 - Null pointer exceptions
 - Undesired side effects
 - Mismatched parentheses in regular expressions
 - ...
- Idea: any **run-time exception** can be detected and prevented **at compile time**
 - Run a more powerful type-checker as a compiler plug-in
 - Warns about potential errors

Two examples of pluggable type-checking

- Fake enumerations
- Units of measure

Fake enumerations

Using `int` where `enum` would have been better

Problems:

- Documentation
- Error checking

Fake enumeration error examples: JabRef bibliography manager

<http://jabref.sourceforge.net>

```
class EntryTypeDialog {
    EntryTypeDialog() {
        // should have been Component.RIGHT_ALIGNMENT (=1)
        component.setAlignmentX(SwingConstants.LEFT); // 2
    }
}

class FileListEditor {
    Thread autoSetLinks(...) {
        // args should be minimum and maximum values
        new JProgressBar(JProgressBar.HORIZONTAL, types.length-1);
    }
}

class FieldNameLabel {
    FieldNameLabel()
        // wrong constant; should be TOP
        label.setVerticalAlignment(NORTH);
    }
}
```

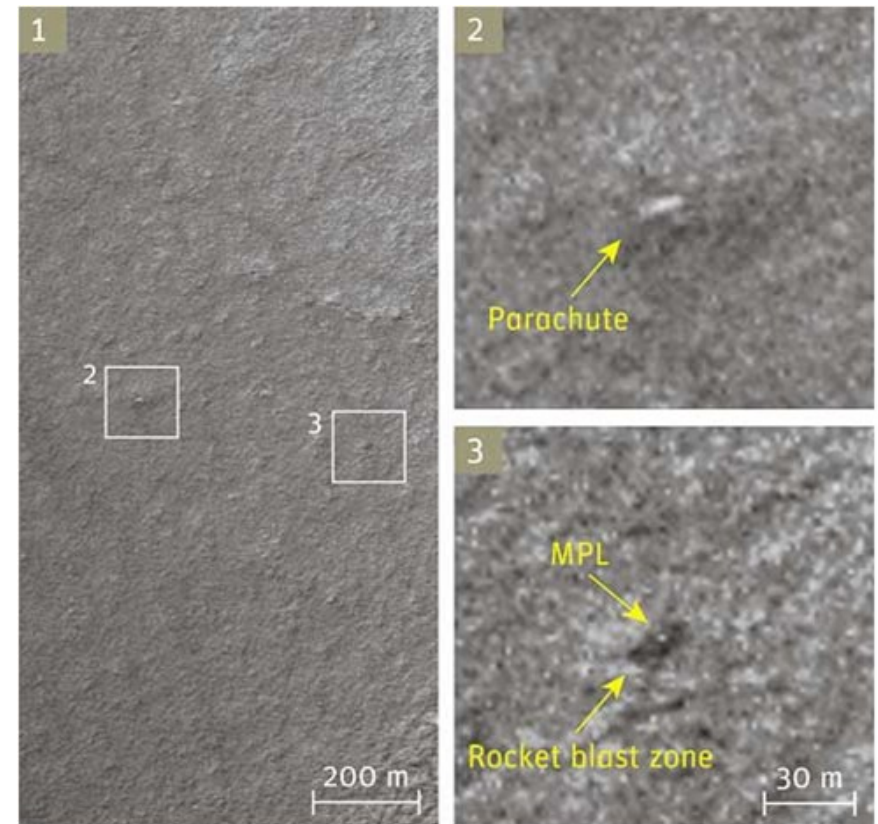
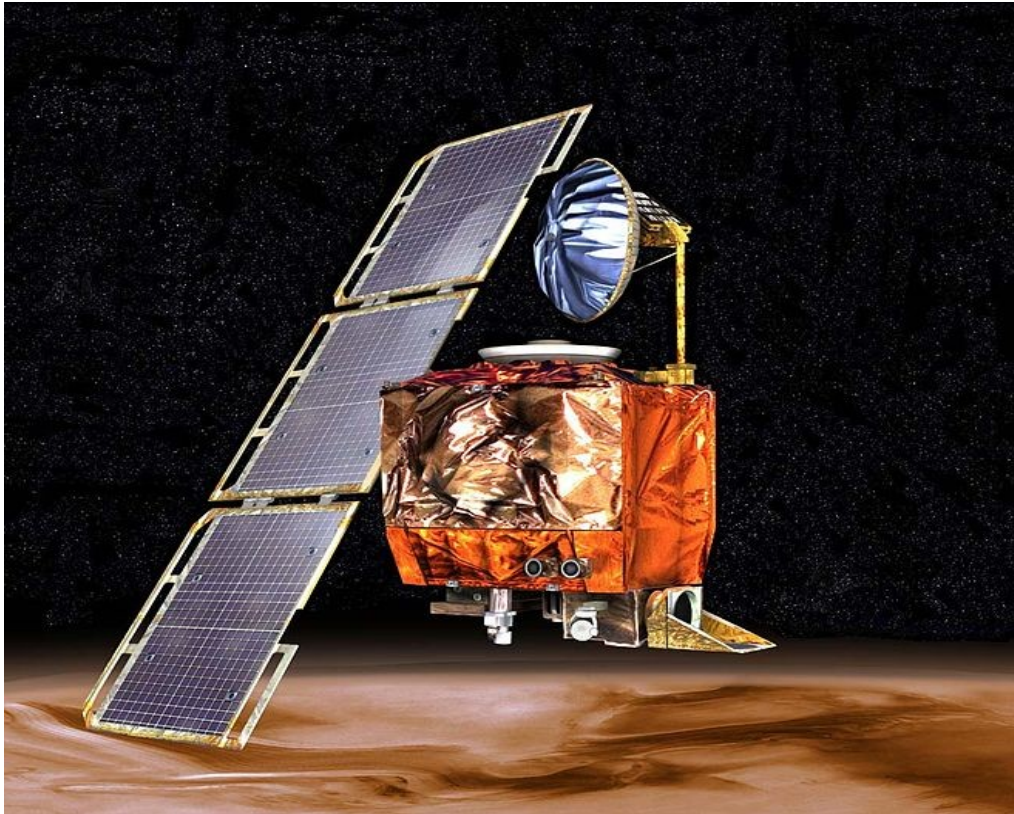
All these bugs have been fixed

Units of measure

- meters vs. yards
- kilograms vs. pounds
- radians vs. degrees

Mars Climate Orbiter

- Unit error in thruster controller: lbf-s vs. N-s
- Crashed into Mars
- 3 years of work, > \$125 million



Gimli Glider

- Fuel computation: pounds instead of kilograms
- Ran out of fuel, glided to an abandoned airfield



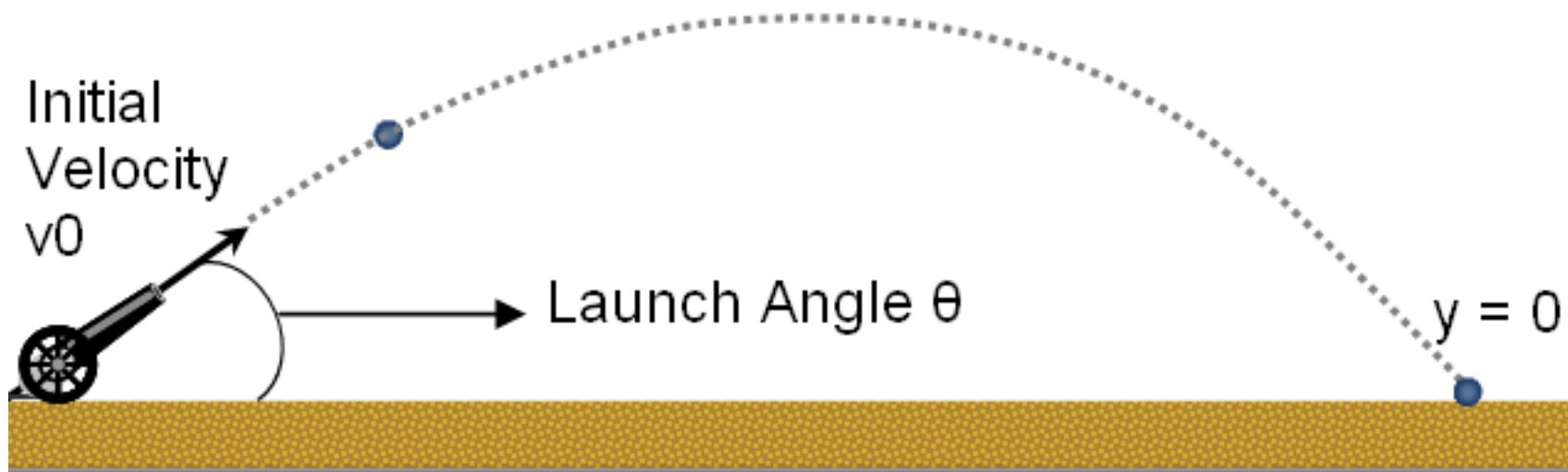
Rampaging turtles

- Moorpark College built an enclosure to contain a 250-pound tortoise
- The first night, 250-kg Clarence pushed over a fencepost and escaped



Ballistics

- How far away do you need to be to be safe?



The Checker Framework

- Tool for pluggable type-checking
 - Comes with many type-checkers
 - null pointers, == vs. equals(), locking, security, string syntax (regex, format strings), internationalization, ...
 - You can build your own
 - Uses Java 8 type annotations (or comments)
 - `List<@Nullable String>`
 - `List</*@Nullable*/ String>`
- Improves documentation, prevents bugs
- Try it! <http://checker-framework.googlecode.com>