Notes - 3/14: SENT

Thursday, July 07, 2011 7:12 PM

Type system:

- bool
- int
- double
- string
- <classes>
- <interfaces>
- <arrays>
- Any?
  - Can ==, ===
- Functions:

DOM:

- IInput input = (IInput)doc.getElementById(“…”)
- Will developers buy into this?
- Or generics:
  - IInput input = doc.getElementById<IInput?(“…”)

Implicit explcit cast:

- Or implicit downcast
- Can assignments and parameters

A world where you use interfaces for external things and classes for internal things is awkward because you have to say things twice.

Decision: Feels like consistency is important – classes and interfaces both support implicit downcasts, classes are enforced at runtime.

Better syntax for lambdas?

Implicit:

- bool -dotdotdot > string
- int -dotdotdot> double
- int -dotdotdot> string
- double -dotdotdot> string

interfaces -dotdotdot> classes

Overloading: Should the result type not be allowed to differ? jQuery does that.

Can envisions a Promise<int> constructor or “int!”

From: Luke Hoban
Sent: Monday, March 14, 2011 9:49 AM
To: Strada Design Team
Subject: Strada Design Meeting Agenda: 3/14

Some possible topics from the design issues backlog:

- Review of metadata encoding (PBCtor)
- Arrays
- Typing of ES5 builtins (Strada core library)
- Optional parameters

- Varargs
- Typeswitch

Others?