Erlang

Jon Bowen

“Erlang is a concurrent language consisting of communicating components where the components are written in a functional language.” – Joe Armstrong

**Key Features**

* Small: 27 reserved keywords
* Process oriented: Programs intended to be organized into discrete processes communicating through message passing
* Functional: Syntax and model highly supportive of functional paradigm
* Concurrent: Concurrent execution built into the language with easy extension to symmetric multiprocessing on multiple cores/CPUs
* Message passing: Processes communicate through message passing in mailboxes (no shared data)
* Immutable data: Variables can only be bound once
* Dynamically typed: No compile time type checking
* Robust: Design patterns and error propagation allow development of extremely robust systems

**Code Samples**

Asdfasd

**Strengths**

* Small language size
* Small set of powerful primitives
* Functional paradigm, immutable data, and message passing reduce side effects and reference errors
* Concurrent/Parallel operation leads to massively scalable and fast systems
* “On-the-fly” code updates
* Error handling model can be used to build extremely robust systems

**Weaknesses**

* Can be slow for large sequential operations compared to procedural languages
* Interpreted rather than natively compiled
* No type checking converts compile time errors to run time errors
* Smaller community of developers
* All or nothing security
* Hurdle of learning the functional language paradigm