

Optimizing Monolithic Compilation in the Google Web Toolkit



Scott Blum
Software Engineer
Google, Inc.



Background

What is Google Web Toolkit?

Whole Program Optimizations in GWT

Future Directions

Q&A

Machines didn't have enough memory to hold the entire program

Separately compiled units

Types might not even be consistent across units

Only local optimizations possible

Linker can perform some pruning, but no feedback loop between global and local optimizations

This model persists into C++ today

Essentially the C model, but with runtime linking instead of compile time

Not even the most basic pruning is done, "Size doesn't matter"

Focus on execution speed driven by dynamic runtime optimizations

Then there's scripting languages



JavaScript

Python

Ruby

...and so on

So why would anyone care about
whole program optimizations?

Google Maps Web Images Groups News Froogle Maps more »

Search the map: Search

e.g., "hotels near lax" or "10 market st, san francisco"

Maps [Print](#) [Email](#) [Link to this page](#)

Map Satellite Hybrid

Y 542 Green St, San Francisco, CA (415) 982-9738

E [Round Table Pizza San Francisco](#)
1503 N Point St, San Francisco, CA (415) 563-1001

F [Pizza Zone](#)
555 Beach St, San Francisco, CA (415) 567-3244

G [Fisherman's Pizzeria](#)
2800 Leavenworth St #50, San Francisco, CA (415) 926-2998

H [Geno's Pizza](#)
421 Beach St, San Francisco, CA (415) 922-0202

I [North Beach Pizza](#)
39 Pier, San Francisco, CA (415) 433-0400

J [Danilo Bakery](#)
516 Green St, San Francisco, CA (415) 989-1806

North Beach Pizza
39 Pier
San Francisco, CA 94133
(415) 433-0400
[northbeachpizza.com](#) - [50 more »](#)
[Send to phone](#)
Directions: [To here](#) - [From here](#)

Golden Gate Nat'l Recreation Area
Fort Mason
Beach St
North Point St

©2006 Google Map data ©2006 NAVTEQ™ Terms of Use

A rich client with interactive UI on the user's browser

Background data fetches with no loss of responsiveness

No installation

UI state is maintained on client

Fewer bits go down the wire

Leverages client CPU and RAM

"No installation" really means "Always reinstall"

Users want web apps to start FAST

Smaller scripts download faster, start quicker, and consume less system memory

Hand-tuning JavaScript for size and performance is a hugely popular aspect of AJAX development

Unfortunately, hand-tuning can turn what started as a beautiful OO design into an unmaintainable pile of hacks

Third-party JavaScript libraries exacerbate the problem

What if a compiler could generate script as good as what you might have hard coded?

Optimizations would make abstraction affordable

Write code according to good software design principles and let the compiler turn it into a hand-tuned pile of garbage

Toolkit for writing AJAX client code in Java

Write, refactor, debug in Java with a Java IDE

Compile to JavaScript + static content

Deploy to any web server

Java's static typing makes possible what is impossible in JavaScript

Compile-time error checking (with an IDE: as you type!)

IDE code completion

Refactor quickly without creating bugs

Perfect obfuscation

Take advantage of whole program optimization

To be continued... 😊